AD-762 720

# DEMONSTRATION OF IMPROVED MONTE CARLO SIMULATION TECHNIQUES FOR THE APAIR ANTISUBMARINE WARFARE PROGRAM


Elgie J. McGrath, et ai

Science Applications, Incorporated


Prepared for:

Office of Naval Research

March 1973


DISTRIBUTED BY:

NTIS

National Technical Information Service
U. S. DEPARTMENT OF COMMERCE
5285 Port Royal Road, Springfield Va. 22151

SAI-72-590-LJ

# DEMONSTRATION OF IMPROVED MONTE CARLO SIMULATION TECHNIQUES FOR THE APAIR ANTISUBMARINE WARFARE PROGRAM

## FINAL REPORT

Scientific Officer, Office of Naval Research (Code 462)

J. R. Simpson

Project Principal Investigator

E. J. McGrath

Co-Authors

D. C. Irving

SCIENCE APPLICATIONS, LA JOLLA, CALIFORNIA
ALBUQUERQUE • ANN ARBOR • ARLINGTON • BOSTON • CHICAGO • HUNTSVILLE • LOS ANGELES
PALO ALTO • ROCKVILLE • SUNNYVALE • TUCSON

P.O. Box 2351, 1250 Prospect Street, La Jolla, California 92037

## ABSTRACT

The use of advanced techniques can greatly improve the effectiveness of Monte Carlo simulation calculations. As a demonstration model, the Navy's Antisubmarine Warfare Air Engagement Model, APAIR, which simulates a single aircraft hunting and destroying a submarine, was selected. Possible improvements in random number generation are presented; however, the study centers on implementation of variance reduction techniques. Two test cases, typical of APAIR implications, were chosen. Examples illustrating the use of the statistical estimation, expected value, systematic sampling, antithetic sampling, correlated sampling, history reanalysis, and importance sampling were run. A comparison of variances with the unmodified APAIR showed the effectiveness of variance reduction techniques. Efficiencies, equivalent to reduced running time to obtain the same variance, of nearly a factor of 20 were obtained in specific cases. It was felt that throung experience and careful effort overall improvements of a factor of 10 could be expected.

Security Classification

**DOCUMENT CONTROL DATA - R & D**

*(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)*

| 1. ORIGINATING ACTIVITY (Corporate author) | 2a. REPORT SECURITY CLASSIFICATION |
|---|---|
| Science Applications, Inc.<br>1200 Prospect Street, La Jolla, California | Unclassified |
| | 2b. GROUP |

**3. REPORT TITLE**

Demonstration of Improved Monte Carlo Simulation Techniques for the APAIR Antisubmarine Warfare Program

**4. DESCRIPTIVE NOTES** (Type of report and inclusive dates)

Final Report

**5. AUTHOR(S)** (First name, middle initial, last name)

Elgie J. McGrath, David C. Irving

| 6. REPORT DATE | 7a. TOTAL NO. OF PAGES | 7b. NO. OF REFS |
|---|---|---|
| March 1973 | ~~87~~ 80 | 11 |

| 8a. CONTRACT OR GRANT NO. | 9a. ORIGINATOR'S REPORT NUMBER(S) |
|---|---|
| N00014-72-C-0293 | |
| b. PROJECT NO. | SAI-73-533-LJ |
| c. NR364-074/1-5-72<br>Code 462 | 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) |
| d. | |

**10. DISTRIBUTION STATEMENT**

Reproduction of this document in whole or in part is permitted for any purpose of the United States Government.

| 11. SUPPLEMENTARY NOTES | 12. SPONSORING MILITARY ACTIVITY |
|---|---|
| | The Office of Naval Research (Code 462)<br>Department of the Navy<br>Arlington, Virginia 22217 |

**13. ABSTRACT**

The use of advanced techniques can greatly improve the effectiveness of Monte Carlo simulation calculations. As a demonstration model, the Navy's Antisubmarine Warfare Air Engagement Model, APAIR, which simulates a single aircraft hunting and destroying a submarine, was selected. Possible improvements in random number generation are presented; however, the study centers on implementation of variance reduction techniques. Two test cases, typical of APAIR applications, were chosen. Examples illustrating the use of the statistical estimation, expected value, systematic sampling, antithetic sampling, correlated sampling, history reanalysis, and importance sampling were run. A comparison of variances with the unmodified APAIR showed the effectiveness of variance reduction techniques. Efficiencies, equivalent to reduced running time to obtain the same variance, of nearly a factor of 20 were obtained in specific cases. It was felt that through experience and careful effort overall improvements of a factor of 10 could be expected.

DD FORM 1 NOV 65 1473

Security Classification

CONTENTS





Preceding page blank

## FIGURES

## TABLES





Preceding page blank

## EXECUTIVE SUMMARY

Contemporary design and construction of large scale Monte Carlo systems analysis programs seldom consider incorporation of the efficient simulation techniques that have been developed, tested and proven successful in various technical disciplines. Most notable among these are random number generation and variance reduction schemes that have been routinely used in radiation transport to provide vast improvements in Monte Carlo simulations.

One objective of a project sponsored at Science Applications, Inc. (SAI) by Code 462 of the Office of Naval Research was to develop these techniques to the point where they would be generally applicable. Basically, this involved development of improved techniques for selecting probability distributions, schemes for generation of random numbers and procedures for variance reduction for general application in Monte Carlo simulation. The results of these developments are presented in Refs. 1, 2, and 3.

Another objective of the project was to demonstrate their applicability to a large scale Navy simulation program.

It is the purpose of this document to summarize the results of the demonstration effort for the Antisubmarine Warfare Air Engagement Model APAIR.[4,5,6] The efficiency estimates showed that significant improvements in APAIR running times (about a factor of 20 in some cases) can be achieved using the improved simulation techniques investigated during this project. It was generally felt that by judicial selection of variance reduction schemes, that running times for the same accuracy can be reduced by a factor of 10 for many APAIR problems.

# 1. INTRODUCTION

The research performed in a project sponsored by the Office of Naval Research (Code 462) at Science Application, Inc. (SAI) over the past year was directed toward achieving the following objectives:

- Develop improved probability selection techniques.
- Develop improved random number generation procedures for selected probability distributions.
- Improve variance reduction technology.
- Demonstrate the application of improved simulation techniques to the ASW air engagement simulation model APAIR.[4, 5, 6]

The results of the effort performed in the first three areas above are documented in Refs. 1, 2 and 3 respectively. It is the purpose of this document to summarize the results of the effort that involved demonstration of improved Monte Carlo Simulation techniques for the APAIR ASW engagement model.

The APAIR model, selected for the study here, is a Monte Carlo simulation of the full engagement between one aircraft and one submarine. The general version (APAIR 2.6) simulates the detection, localization and attack phases of airborne ASW missions. The model is particularly suited for this study since it is a relatively long running program, and because it is in rather wide use, improvements in the running time would be most welcome. It is emphasized, however, that the objective was not to critique the APAIR model but rather to evaluate the effectiveness of efficient techniques which are not commonly used in Monte Carlo simulations. It should also be mentioned that not only are the techniques studied here applicable to the APAIR program but also to other ASW simulation models such as APSURV,[7] APSUB and APSURF.[8]

The version of APAIR used in the study was provided to SAI by Code 141, Naval Undersea R/D Center (NUC) in San Diego. In addition to the program, Code 141 also provided SAI with a "representative" set of data for demonstration purposes only. No attempt was made to interpret any of the results beyond that necessary to evaluate the improved simulation efficiency. However, the results are presented in sufficient detail that this document should be generally useful to the analyst interested in improving the random number generation schemes and incorporation of variance reduction in APAIR.

Of the three areas considered for improvement in APAIR (probability distribution selection, random number generation and variance reduction), variance reduction proved to be the most successful. Improvement in the probability distribution modeling was restricted due to the lack of sufficient data on input parameters (such as aircraft navigation errors, sonobuoy fixes and drop errors, aircraft weapons effects, tactics, etc.).

In the area of random number generation, several modifications were identified which would not only improve the generation of random numbers but also increase the ease with which random numbers could be generated on various computers. The latter was accomplished by developing a random number generator that was machine independent. This is particularly important when comparison between results from different machines or facilities are desired or when the program is to be made operational at new facilities.

The third area involved the application of variance reduction techniques. This proved to be extremely fruitful. The variance reduction techniques used here were developed primarily for application to radiation transport problems and have not been widely used. However, it was shown conclusively here that a much broader application is warranted. For example, in some cases improvements in efficiency (i.e., the running time required to achieve the

same variance) was a factor of almost 20. An overall improvement of a factor of 10 could generally be expected if the effort to understand and carefully apply the various techniques is expended. A summary of the improvement in efficiency in terms of running time reduction is shown in Table 1.1 for the various variance reduction techniques applied to APAIR.

In the following section of the report the improvements in random number generation are described. The use of variance reduction in APAIR will be described in detail in Section 3.

Since the APAIR study was for demonstration purposes only, it should be mentioned that only expedient modifications were made to the program. Therefore, the APAIR program at SAI containing the improvements discussed is not considered to be a version that would be generally useful. However, it has been proposed that these modifications be made to APAIR to provide for user flexibility and improved efficiency in the future.

# TABLE 1.1

## A Summary of Efficiency Improvements in APAIR Application Using Variance Reduction Techniques

| ASW Parameters Considered For Variance Reduction | Efficiency of Simulation With Variance Reduction Technique Applied* | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Statistical Estimation (1st Application) | Statistical Estimation (2nd Application) | Expected Value for 2nd Torpedo (1st Application) | Expected Value for 3rd Torpedo (1st Application) | Expected Value for 2nd Torpedo (2nd Application) | Expected Value for 3rd Torpedo (2nd Application) | Systematic Sampling (1st Application) | Systematic Sampling (2nd Application) | Antithetic Variates (1st Application) | Antithetic Variates (2nd Application) | Correlated Sampling | [illegible] | Importance Sampling |
| Probability of Submarine Kill | 1.33 | 1.09 | 2.36 | 10 | 2.00 | 10.7 | 0.96 | .92 | 1.09 | 0.92 | 3.0 | 1.21 | 6.5 |
| Time to Submarine Kill | 1.19 | 1.39 | 1.09 | -- | 1.74 | 17.1 | 1.07 | 1.08 | 1.14 | 1.11 | 1.58 | 7.2 | 1.5 |
| Number of Torpedos Used | 1.25 | 1.22 | -- | -- | -- | -- | 1.78 | .67 | 1.31 | 1.36 | 1.33 | 9.3 | 19.7 |
| Number of Sonobuoys Used - Type A | 1.23 | 1.14 | 1.21 | -- | 1.62 | 15.4 | 0.71 | 1.38 | 1.00 | 1.32 | 2.40 | 6.4 | 3.0 |
| Number of Sonobuoys Used - Type B | 1.13 | 1.50 | 1.48 | -- | 1.37 | 10.2 | 2.19 | 2.14 | 1.17 | 2.14 | 1.17 | 9.7 | 5.0 |
| Average Improvement in Efficiency** | 1.15 | 1.23 | 1.51 | 10 | 1.69 | 14.1 | 1.34 | 1.28 | 1.12 | 1.37 | 1.99 | 5.8 | 7.2 |

*The efficiency can be interpreted as the factor with which the running time can be reduced by using the corresponding variance reduction technique indicated to achieve the same variance as when no variance reduction techniques were used.

**This is simply a numerical average of the numbers in the column above.

## 2. RANDOM NUMBER GENERATION IMPROVEMENTS IN APAIR

In this section, the potential improvements in the APAIR random number generators are discussed. It is not anticipated that replacement of the random number generators currently used in APAIR would significantly reduce the total program running time in most cases, since the fraction of time spent in generating random numbers in a typical application is not large. However, in some cases, the approaches recommended here will be faster, more accurate and more convenient.

Possibly the most significant improvement for APAIR random number generation would be in the use of a machine independent uniform random number generator. This will automatically permit identical sequences of random numbers to be generated on different computers. The main advantage in this, of course, is that identical results can be obtained on different machines and at different installations for comparison purposes. Furthermore, since the existing random number generator now used in APAIR is machine dependent, transferring the code to other machines can cause considerable difficulty. Use of the machine independent version described here would eliminate this problem. The use of the machine independent uniform random number generator and improvements in the normal distribution will be discussed below.

### 2.1 UNIFORM RANDOM NUMBER GENERATOR

The uniform random number generator in the current version of APAIR obtained by SAI from NUC and designed for use on the Univac 1108 is a mixed congruential generator employing the algorithm

$$X_{n+1} = 27095269935 \cdot X_n + 2049 \pmod{2^{35}}$$

where $X_n$ and $X_{n+1}$ are successive random numbers. This generator has not been subjected to the Coveyou-MacPherson analysis (Ref. 9), the most

exacting test for random number generators currently known. Therefore, its validity as a random number generator is not proven. There are, however, no a priori reasons to suspect that this generator is faulty.

This generator is, however, not machine independent. It will work only on the Univac 1108 and other machines with a 36-bit word length and a similar negative integer representation. It will not work, for example, on an IBM 360 computer. A better choice of a basic random number generator would be the machine independent generator, MIRAN, described in Appendix A. This would allow greater flexibility in exchange of program and problems between different computer facilities and in addition uses an algorithm of proven validity. Random number generation occupies such a small portion of the overall APAIR computing time that the loss of efficiency entailed in using MIRAN would not cause an increase in the total running time of APAIR problems.

## 2.2 NORMAL RANDOM NUMBER GENERATOR

The normal distribution is used in APAIR to generate random variables such as aircraft navigation errors (inertial, tactical, navigation, reset, areas of uncertainty for sonobuoy fixes, etc.). The APAIR procedure currently used to obtain a normal random variable from the distribution:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2.1}$$

is to generate an approximate normal random deviate using

$$X = \sigma\left(\sum_{i=1}^{12} R_{u_i} - 6\right) + \mu \quad . \tag{2.2}$$

Where $R_{u_1}, \ldots, R_{u_{12}}$ is a set of 12 independent random values from the uniform distribution $U(0,1)$. This procedure takes about 105 microseconds on a UNIVAC 1108 using assembly language.

The result used in (2.2) is based on the central limit theorem[10] and is, therefore, approximate. For example, the range of X using (2.2) is limited to

$$\mu - 6\sigma \leq X \leq \mu + 6\sigma \tag{2.3}$$

which is probably adequate in most situations. Difficulty could arise when very small probability events (i.e., outside the range of X as given by (2.3)) are important.

A better method for generating random numbers from the normal distribution which is both exact (within machine accuracy) and requires only 30 microseconds is a technique developed by Marsaglia[11] and documented in Ref. 2. A routine using this approach, is shown below along with the corresponding flow diagram in Fig. 2.1.

$$f(x) = (2\pi)^{-1/2}\, e^{-x^2/2} \quad ; \quad -\infty < x < \infty$$

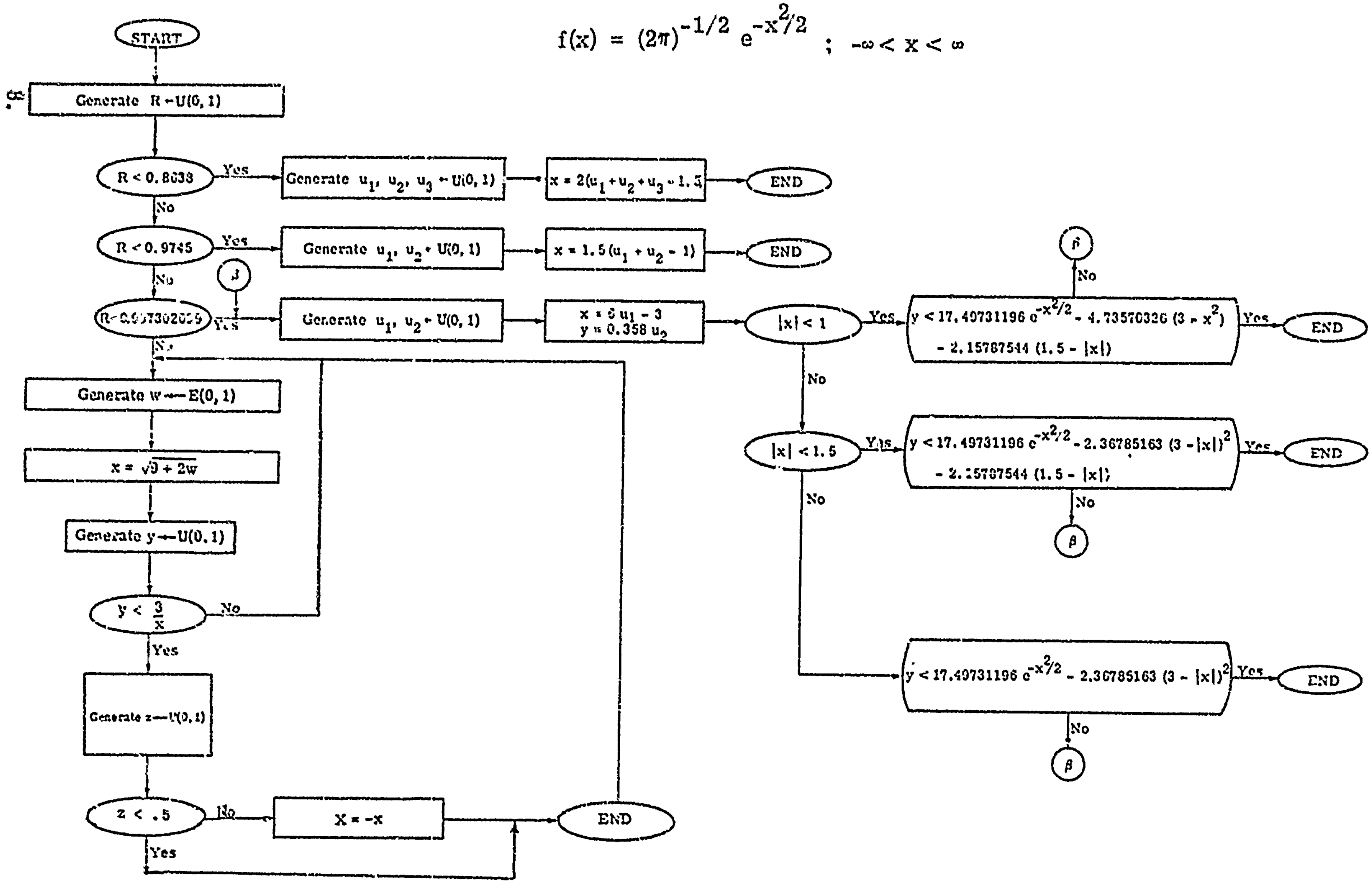



Figure 2.1. Normal Distribution

Sample Routine

```
          FUNCTION RANORM (DUMMY)
          R = RANUMB(R)
          IF (R. GT.0. 8638) GO TO 10
          RANORM = 2. *(RANUMB(X) + RANUMB(Y) + RANUMB(Z) - 1. 5)
          RETURN
   10     IF (R. GT.0. 9745) GO TO 20
          RANORM = 1. 5*(RANUMB(X) + RANUMB(Y) - 1. 0)
          RETURN
   20     IF (R. GT.0. 997302039) GO TO 100
   25     X = 6. *RANUMB(X) - 3. 0
          Y = 0. 358*RANUMB(X)
          XSQ = X*X
          GX = 17. 49731196*EXP(-XSQ*. 5)
          AX - ABS(X)
          IF (AX. GT. 1. 1) GO TO 30
          IF (Y. GT.(GX-17. 44392294 + 4. 73570326*XSQ + 2. 157987544*AX))
          GO TO 25
          RANORM = X
          RETURN
   30     AX3 = 2. 367985163*(3-AX)**2
          IF (AX. GT. 1. 5) GO TO 40
          IF (Y. GT.(GX-AX3-2. 157987544*(1. 5-AX))) GO TO 25
          RANORM = X
          RETURN
   40     IF (Y. GT.(GX-AX3)) GO TO 25
          RANORM = X
          RETURN
  100     X = SQRT (9-2*(ALOG(RANUMB(X)))
          IF (RANUMB(X). GT. 3/X) GO TO 100
          IF (RANUMB(X). GT. 0. 5) X = -X
          RANORM = X
          RETURN
          END
```

As written above, the routine generates a normal standard deviate (i. e., a random number from (2. 1) with $\sigma = 1$ and $\mu = 0$) defined as N(0, 1). It is then left up to the calling program to multiply by the standard deviation and add the mean if a generalized normal deviate is required. That is, for a distribution with mean $\mu$ and variance $\sigma^2$, (i. e., Eq. 2. 1) the correct

random number would be $\sigma N(0,1) + \mu$, where $N(0,1)$ is a random number from a distribution with $\mu = 0$ and $\sigma^2 = 1$.

Therefore, if the algorithm prescribed above is used in APAIR, the running time for generation of random numbers from a normal distribution would improve almost a factor of 4 and it also will provide exact answers.

## 3. VARIANCE REDUCTION IMPROVEMENTS IN APAIR

The type of problems solved by APAIR provide an excellent opportunity to apply a wide variety of variance reduction techniques which can substantially improve APAIR efficiency. Several of these were tried on APAIR with generally excellent results. In some applications the gain in efficiency was estimated to be almost a factor of 20. It appears that a factor of 10 improvement can readily be accomplished in many APAIR applications if the time and effort is expended to understand and implement the appropriate techniques. It is unquestionably to the benefit of the analyst to follow such an approach when long running times are of concern.

Resources available to perform the study did not permit application of all the possible variance reduction schemes available. Therefore, a selected number of techniques representing a broad range of characteristics were applied to various types of ASW problems. These included:

- Statistical Estimation using an expected value of kill at each torpedo drop rather than scoring actual kills based on Monte Carlo simulation. Two cases were studied corresponding to different types of Magnetic Anomaly Detectors (MAD).

- Expected Value replacing actual submarine kill by a reduction in the "survival value" of the submarine for computing percent kill as a function of which torpedo caused the kill (first, second, etc.). This was also performed for two types of MAD gear.

- Systematic Sampling selecting initial starting coordinates for the submarine location using two types of MAD gear. Two types of systematic sampling were considered.

- Antithetic Variates used to select initial starting locations of the submarine.

- Correlated Sampling using random number control to generate identical histories to a point where differences in two types of MAD gear lead to a divergence of histories. This was used to estimate differences in effectiveness between two types of MAD gear.

- History Reanalysis where one run was made unbiased and the second generated using weighting factors on the histories from the first to correct for differences in MAD gear. This was also used for estimating differences in effectiveness between two types of MAD gear.

- Importance Sampling performed with an importance function weighted to generate correlated samples for the two types of MAD gear. Again, this was used for estimating differences in effectiveness between two types of MAD gear.

As a basis for quantitatively characterizing the calculational efficiency of variance reduction over straightforward or crude sampling (i.e., with no variance reduction) the following definition was used throughout

$$\epsilon = \left(\frac{\text{variance with crude sampling}}{\text{variance with the variance reduction technique}}\right) \times \left(\frac{\text{APAIR running time with crude sampling}}{\text{APAIR running time with variance reduction}}\right)$$

Crude sampling represents the procedure currently used in APAIR. Using the above definition for efficiency, a value of $\epsilon = 2$ for application of a variance reduction technique implies a reduction in computer time by 1/2 to achieve the same variance as would be obtained with crude sampling. The rationale for using this as an efficiency factor is discussed in Ref. 3.

It should be recognized that the efficiency, $\epsilon$, as defined above is a random variable since the variances with and without variance reduction are estimates generated from the statistics of the simulation and not theoretical analyses. A batching method was used in estimating most of these variances. The procedure is presented in Ref. 3 and will not be detailed here. It is important to recognize, however, that the efficiencies presented are estimates and, therefore, are subject to a certain variability. In fact, the variance estimates are second order quantities and thus this variability is generally much greater than that encountered for the primary quantity whose variance is being estimated.

From theoretical arguments the error in the efficiency figure could easily be as large as $\pm$ 45%. In some of the cases presented in this report, there are large correlations between the crude simulation and the variance reduced technique and thus the variability of the efficiency figure is much less than this theoretical maximum. Since efficiency is so difficult to calculate with any accuracy, calculations were made for five similar parameters—kill probability, mean time to kill, number of stores used per kill (for three types of stores: torpedoes and two kinds of sonobuoys). In general the variance reduction efficiencies for these parameters should not be greatly different, therefore, the spread in efficiency values for these five parameters should give a rough idea of the variability of the efficiency figure for each variance reduction technique. As an improved estimate of the efficiency of each technique, a simple average of the five efficiency figures is presented in the tables.

The initial variance reduction technique implemented was statistical estimation. Implementation involved patching into the report generator portions of the program to call subroutines designed to score results using statistical estimation. These routines contained the batching needed for estimation of variances, print statements for the results, etc. To minimize the programming changes needed to implement subsequent techniques, the statistical estimation subroutines were left in the program to calculate and print results. Thus each of the subsequent comparisons that were done actually contrasted a variance reduction technique plus statistical estimation to a case with only statistical estimation. However, with the exception of the expected value technique as discussed in Section 3.3, it was felt that the efficiency factors obtained were essentially the same as if the variance reduction technique alone had been compared to crude Monte Carlo.

Each of the variance reduction techniques used here have been described in detail in Ref. 3 and will not be presented to that extent in this report.

However, in the following discussions, the techniques and the results of the application of the techniques will be described in sufficient detail to provide an appreciation for the steps involved. Before proceeding, a brief description of the general type of APAIR problem considered in the study will be presented.

## 3.1 APAIR PROBLEM DESCRIPTION

APAIR can be applied to a variety of ASW problems involving one aircraft in pursuit of a submarine. The problem addressed here is considered to be a typical application of APAIR in that it is designed to estimate the effectiveness of the aircraft and sensors in finding and killing a submarine.

A total simulation of APAIR is comprised of a series of independent histories. A single history in the simulation proceeds as follows:

The initial aircraft location, direction and speed are chosen. The submarine location, bearing and speed are also chosen according to specified random characteristics. The aircraft proceeds to execute certain tactics with the objective of detecting the submarine.

Once the submarine is detected, the aircraft enters a localization phase where certain maneuvers are performed and sonobuoys are dropped in a specified pattern. In addition, Magnetic Anomaly Detection (MAD) gear is used in the localization process. Two typical problems, differing only in the range of detection of the MAD gear, were utilized in this study.

If localization of the submarine has been achieved, the aircraft proceeds to drop a torpedo which may or may not result in a kill. If no kill is realized, the aircraft then proceeds through further localization to attempt another torpedo drop, again following specified maneuvers. The history can be terminated in several ways which include exceeding a time limit, achieving a submarine kill, or exhausting aircraft stores (torpedoes or sonobuoys).

Once the history is terminated, by any event whatever, a new history is initiated. When the specified number of histories are completed, final results are tabulated and printed out. Among the parameters estimated by APAIR the following were used in this study to illustrate the efficiency of variance reduction:

- Probability of a submarine kill
- Average time used to achieve a submarine kill
- Number of stores used per submarine kill for three types of stores. One type was torpedoes and the other two were two kinds of sonobuoys. (As the exact identification of the sonobuoy types was not clear from our documentation and irrelevant to this study, they are merely labelled type A and type B in this report. A third type of sonobuoy was, for the tactics in our sample problem, not a random variable and thus was not included)

Improvement in the estimate of these basic parameters was the objective in using the variance reduction techniques considered here. Of course, such an improvement can also be achieved by increasing the number of histories, although the running time can become prohibitively long. For example, the above problem required approximately 15 minutes on a Univac 1108 to generate 100 histories. The desirability for achieving variance reduction is therefore obvious.

## 3.2 APPLICATION OF STATISTICAL ESTIMATION

In the course of an individual history, the aircraft will drop a torpedo when it thinks it has determined the submarine's location and heading. The actual submarine speed, aspect, and range are used to determine, from input tables, the probability, $p_K$, that the torpedo will destroy the submarine. A random number, $R_u$, from a uniform distribution is generated and compared to $p_K$. If $R_u < p_K$, a kill is scored and the history is terminated. If $R_u \geq p_K$, no kill occurs and no scoring is done; the game continues with more localization and possibly, another attack. If other torpedoes are dropped, a similar procedure is used to determine if a kill is scored later in the game.

At the completion of the simulation, the probability of submarine kill is estimated as

$$\hat{p}_K = n/N$$

where

$n$ = number of kills scored and

$N$ = number of histories run.

For estimating the time to kill, APAIR currently uses

$$\hat{T}_K = 1/n \sum_{i=1}^{n} T_{Ki}$$

where $n$ is the number of kills scored and $T_{Ki}$ ; $i = 1, \ldots, n$ the time taken to effect a kill in the $i^{th}$ history which ended in a kill. Similar procedures are used for the other parameters (number of torpedoes and sonobuoys used per kill) estimated.

In the statistical estimation technique for variance reduction, the scoring was changed so that actual kills are not scored, but rather the expected value of kill, $p_K$, was scored for all torpedo drops regardless of whether or not a kill was achieved. However, the simulation game itself was not modified in any way. That is, a random number, $R_u$, was still used at each torpedo drop to determine whether the history was terminated by a kill or more maneuvers took place. The outcome of this random choice did not affect the scoring of $p_K$.

Specifically, define

$p_{Kij}$ = probability of kill for the $j^{th}$ torpedo dropped during history $i$.

$n_i$ = number of times a torpedo was dropped during history $i$.

Then, the total score generated for history $i$ is

$$\sum_{j=1}^{n_i} p_{Kij} \quad ; \quad i = 1, \ldots, N$$

and the estimate for the probability of kill for the entire simulation (N histories) is

$$\hat{p}_K = 1/N \sum_{i=1}^{N} \sum_{j=1}^{n_i} p_{Kij}$$

In the case of the remaining parameters, a similar approach was taken. For example if

$T_{Kij}$ = time to torpedo detonation for the $j^{th}$ torpedo dropped in history $i$

then the estimate for time to kill is given by

$$\hat{T}_K = \frac{1/N \sum_{i=1}^{N} \sum_{j=1}^{n_i} p_{Kij} T_{Kij}}{\hat{p}_K}$$

Similar expressions were used for the remainder of the parameters being estimated.

The computational time per history is very slightly increased using this technique since the same game is still being played but there is a little more bookkeeping in the scoring. However, this is offset by the resulting variance reduction.

In demonstrating the statistical estimation technique, both the crude Monte Carlo estimate (counting actual kills) and the statistical estimate (summing over $p_K$ values) were calculated in the same run and, therefore, used the same histories. This produces a high degree of correlation between the crude and the statistical estimation results which reduces the variance of the efficiency figure. Two problems were run using MAD gear having long and short range detection capabilities respectively. The variances obtained with statistical estimation and with crude Monte Carlo are shown in Tables 3.1 and 3.2 along with the resulting efficiency factor for the use of this variance reduction technique. The sample variances were estimated using the statistical techniques described in Ref. 3 to obtain the variance results indicated. The actual estimated values of the parameters are not presented since they are not considered to be germane to comparison of the efficiencies. Therefore, only the variances in the two cases are shown along with the efficiencies obtained in estimating the parameters with variance reduction.

The efficiencies obtained varied from 1.00 (implying no improvement) to as high as 1.50 (implying a factor of 1.5 reduction in running time). However, these extreme values probably represent statistical fluctuations in the variance

TABLE 3.1

Variance Reduction Using the Statistical Estimation Technique For the Short Range MAD Gear

| Estimated Expected Values | Variance With Crude Sampling | Variance With Statistical Estimation | Efficiency of Variance Reduction |
|---|---|---|---|
| Probability of submarine Kill | $2.5 \times 10^{-3}$ | $2.4 \times 10^{-3}$ | 1.03 |
| Time to submarine Kill | 10.2 | 9.15 | 1.10 |
| Number of torpedos used per Kill | $2.39 \times 10^{-3}$ | $1.9 \times 10^{-3}$ | 1.25 |
| Number of Buoys Used per Kill - Type B | 0.084 | 0.067 | 1.23 |
| Number of Buoys Used per Kill - Type B | 0.450 | 0.396 | 1.13 |

Average efficiency improvement $\cong$ 1.15

Ratio of running times (increase with variance reduction) $\cong$ 1.01

Statistical Estimation. Rather than score actual kills, an expected value of kill is scored at each torpedo drop regardless of whether or not the torpedo scored a kill. The game is not modified.

TABLE 3.2

Variance Reduction Using the Statistical Estimation Technique For the Long Range MAD Gear

| Estimated Expected Values | Variance With Crude Sampling | Variance With Statistical Estimation | Efficiency of Variance Reduction |
|---|---|---|---|
| Probability of submarine Kill | $2.5 \times 10^{-3}$ | $2.5 \times 10^{-3}$ | 1.00 |
| Time to sub-marine Kill | 9.20 | 7.00 | 1.30 |
| Number of Torpedos used per Kill | $6.1 \times 10^{-3}$ | $4.96 \times 10^{-3}$ | 1.22 |
| Number of Buoys Used per Kill - Type A | 0.076 | 0.066 | 1.14 |
| Number of Buoys Used per Kill - Type B | 0.50 | 0.33 | 1.50 |

Average Efficiency Improvement $\cong 1.23$

Ratio of time increase with variance reduction $\cong 1.01$

Statistical Estimation: Rather than score actual kills, an expected value of kill is scored at each torpedo drop regardless of whether or not the torpedo scored a kill. The game is not modified.

estimates rather than real efficiency improvement. Average efficiencies of 1.15 for the short MAD problem and 1.23 for the long were achieved. Thus, it can be seen that use of statistical estimation could have made roughly a 19% improvement in APAIR run times. Use of statistical estimation is justified since it involves a very trivial modification to the program.

### 3.3 EXPECTED VALUE TECHNIQUE FOR ESTIMATING EFFECTIVENESS OF MULTIPLE TORPEDO DROPS

The expected value technique differs from statistical estimation in that the actual game or simulation being played, as opposed to just the scoring technique, is changed to replace a random choice with an expected value for the outcome of that choice. For example, instead of generating $R_u$ to test against $p_K$ with the choice of either killing or missing the submarine, the submarine is given an initial "weight" of 1.0. If the first torpedo dropped has a $p_K$ of .80, then 80% of the submarine is deemed killed but 20% survives and the weight is reduced to .2. If a second torpedo is dropped which has a $p_K$ of .50, then half of the remaining weight, or .1, is killed, while a weight of .1 continues to survive. The history is never ended due to a submarine kill but continues until some other limit, such as using up the mission time or using up the stores of torpedoes or sonobuoys, stops the history.

Such a technique could be useful in studies such as determining the effectiveness of the number of torpedoes carried on an aircraft or the worthiness of the tactics for relocalization and reattack following a miss by the first torpedo. In these cases one is not so much interested in the overall kill probability as in the kills scored by the second, third, etc. torpedoes. In a crude Monte Carlo most of the kills are made by the first torpedo and the history ends there. These histories add nothing to the knowledge of tactics involving reattack or to the kill value of the second torpedo, but simply constitute wasted time towards calculating the items of importance. In fact, what is worse, these histories add variance to the overall kill probability. It would

be prohibitively expensive to determine the value of reattacks from a crude Monte Carlo due to the large proportion of the running time being spent on histories of no value to this parameter.

Using the expected value technique outlined above, most histories will contribute to knowledge concerning the second and third torpedoes as the history will always continue after the first torpedo drop. For example, one of the crude Monte Carlo problems run had (out of 100 histories) 78 cases in which the first torpedo was dropped, 10 in which the second was dropped, and only one history where the third torpedo was dropped. Using the expected value technique, the same problem had 79 histories with one torpedo drop, 65 histories in which the second torpedo was dropped, and 43 histories of a third torpedo drop.

Estimation in the expected value case was done as follows. If $p_{Kij}$ is the kill probability for the $j^{th}$ torpedo in the $i^{th}$ history and $w_{ij}$ is the submarine weight at the time that torpedo was dropped, then

$$\hat{p}_{Kj} = 1/N \sum_{i=1}^{N} p_{Kij} w_{ij}$$

is the kill probability of the $j^{th}$ torpedo. Likewise, the time to kill by the $j^{th}$ torpedo is given by

$$\hat{T}_{Kj} = \frac{1}{\hat{p}_{Kj}} \sum_{i=1}^{N} T_{Kij} p_{Kij} w_{ij}$$

where $T_{Kij}$ is the time of detonation of the $j^{th}$ torpedo on the $i^{th}$ history. A similar formula is used to estimate the numbers of sonobuoys dropped per kill by the $j^{th}$ torpedo. (Obviously the number of torpedoes used is constant in this case, so this parameter was omitted.)

For this technique, two sample problems were run; one with the long-range MAD gear and one with the short-range MAD gear. In each case APAIR was run twice, with and without the expected value technique. To reduce the variance of the efficiency factors, the histories in the two runs were correlated using the technique described in Section 3.6. This kept the histories in each run identical through the first torpedo drop. The resulting efficiencies for the second and third torpedo drops for both cases are shown in Table 3.3. There were no examples of a fourth torpedo drop in any of the runs. In the short MAD case, there were no examples of a third torpedo drop in the crude Monte Carlo run, so the efficiency of the variance reduction technique is theoretically infinite in this case. However, using the $\hat{p}_K$ estimate from the expected value run, it was possible to estimate the number of crude Monte Carlo histories that would be necessary to get similar statistics; this led to the efficiency factor of 10 for $\hat{p}_K$ shown for the third torpedo in the short MAD case.

The running times for the crude and expected value calculations are shown in Table 3.4. As anticipated, the expected value histories took much longer to run because they did not stop at the first torpedo drop (as most of the crude Monte Carlo histories did) but went on to simulate a second and a third torpedo drop. This extra running time is used to generate information concerning the parameters of interest, i.e., the kills made by second and third torpedo drops, and, therefore, the overall efficiency is much improved as Table 3.3 shows. It is instructive to consider the efficiency for the first torpedo drop. As the histories in the two runs were identical through the first torpedo drop, the variances are identical for the first torpedo parameters. Due to the increased running time, the first torpedo efficiencies will be lower by .49 (for the short MAD) and .36 (for the long MAD). The extra running time used in calculating second and third torpedo drops is wasted as far as first torpedo drops is considered and this lowers the efficiency. This illustrates a common point of variance reduction: any technique which reduces variance

TABLE 3.3

Efficiency Calculated for Expected Value Technique

| Estimated Parameter | Problem 1 Long Range MAD Gear | | Problem 2 Short Range MAD Gear | |
|---|---|---|---|---|
| | 2nd Torpedo | 3rd Torpedo | 2nd Torpedo | 3rd Torpedo |
| Probability of Kill | 2.00 | 10.7 | 2.36 | *10 |
| Time to Kill | 1.74 | 17.1 | 1.00 | -- |
| Sonobuoys Used per Kill - Type A | 1.63 | 18.4 | 1.21 | -- |
| Sonobuoys Used per Kill - Type B | 1.37 | 10.2 | 1.46 | -- |
| Average Efficiency | 1.69 | 14.1 | 1.51 | *10 |

Efficiency factors calculated from a comparison of expected value to statistical estimation have been increased by 20% to make comparison of expected value technique with crude Monte Carlo.

*Theoretical estimate. Actual estimate was infinite as crude Monte Carlo did not contain any realizations of a third torpedo drop for this case.

Expected value: At each torpedo drop, the 'weight' of the submarine is decreased by the torpedo kill probability. No random game to determine kill/miss is played and the simulation continues.

TABLE 3.4

Running Times for Crude Monte Carlo and Expected Value Calculations

| Problem | Running Time for 'Crude' Monte Carlo | Running Time for Expected Value Technique | Ratio of Times |
|---|---|---|---|
| Problem 1: Long MAD | 664 sec | 1866 sec | 2.81 |
| Problem 2: Short MAD | 822 sec | 1666 sec | 2.03 |

for one parameter will increase variance for some other parameter. Variance reduction techniques must be carefully tailored to the parameters of importance, in this case the kills involving two and three torpedo drops.

For ease in making programming changes to calculate the parameters separately by torpedo drop, base runs with the current APAIR, representing crude Monte Carlo, were not made. Runs using the expected value technique then provided efficiency factors for an expected value/statistical estimation comparison. The average efficiency of the statistical estimation/crude Monte Carlo comparison in Section 3.2 was determined to be a factor of 1.2. Multiplying the efficiencies from the expected value/statistical estimation comparisons by 1.2 generated the figures presented in Table 3.3 as the efficiency of expected value versus crude Monte Carlo. This combination of efficiency factors is justified because the expected value technique necessarily incorporates statistical estimation scoring; once the kill/miss decision has been removed from the game, the scoring must be by the $p_K$'s for each torpedo drop. Thus there is no possible "expected value without statistical estimation" comparison to crude Monte Carlo but only the combined efficiency.

## 3.4 SYSTEMATIC SAMPLING OF INITIAL SUBMARINE POSITION

Systematic sampling is a variance reduction technique that usually finds application in selecting initial or starting values for a random variable. Potential applications in APAIR include initial submarine or aircraft bearing and location. Sampling in a systematic manner essentially serves to reduce the contribution to the variance coming from the random variables being systematically sampled.

To demonstrate this technique in APAIR, problems were run with the aircraft initial location and both the aircraft and submarine initial bearings fixed. This left the submarine starting location as the variable which was systematically sampled as shown in Fig. 3.1. The aircraft was initially located at the origin while the submarine starting position was uniformly distributed along the y-axis (north) between 0 and L. Both the aircraft and the submarine were initially moving east as shown.

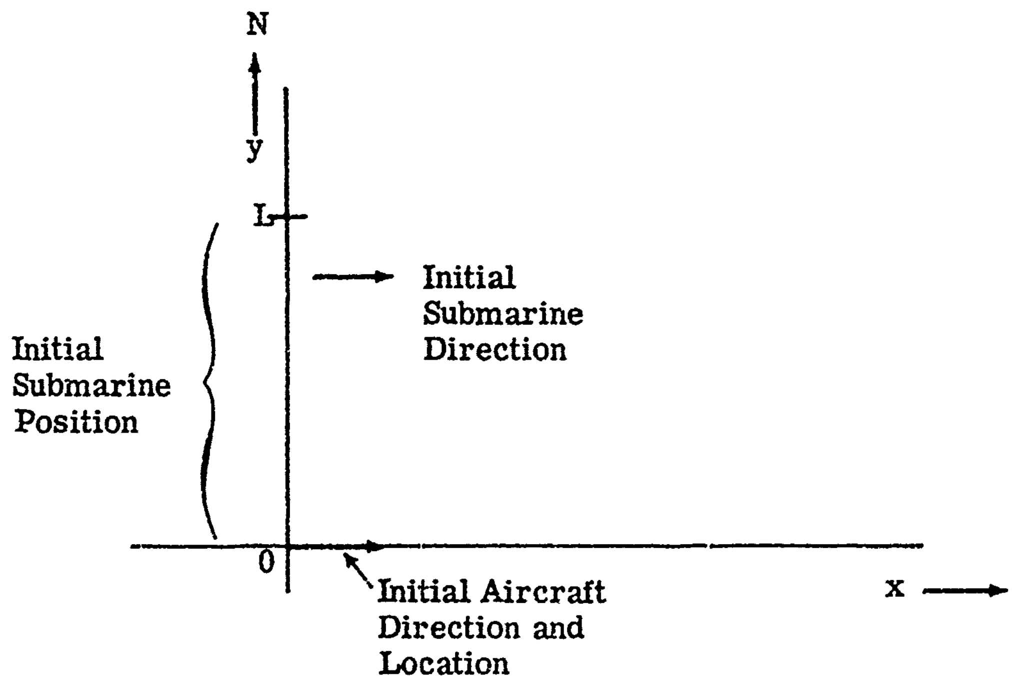



Fig. 3.1. Starting Positions for Systematic Sampling Demonstration

Systematic sampling was applied in two different ways. The first was used on the long MAD gear problem and was implemented as follows:

To obtain starting positions for the first 10 histories, a random number $R_u$ was selected from U(0, 1) and ten initial submarine positions were located at

$$y_1 = L\,R_u/10$$

$$y_2 = L/10 + LR_u/10$$

$$y_3 = 2L/10 + LR_u/10$$

$$\vdots$$

$$y_{10} = 9L/10 + LR_u/10$$

The histories run using these initial starting conditions constituted the first batch. Then another series of ten y's was generated by selecting a second random number from U(0, 1), and the second batch was run. The process was continued until a total of 10 batches of results (or 100 histories) were

obtained. The results of these simulations were used to estimate the parameters and sample variances using batched estimators. That is, an estimate for $p_K$ was obtained for each batch by the usual methods. These batch estimates may be denoted $\hat{p}_{K1}, \cdots, \hat{p}_{K10}$. A final estimate was obtained from

$$\hat{p}_K = 1/10 \sum_{i=1}^{10} \hat{p}_{Ki}$$

and the sample variance was estimated from

$$\hat{S}^2 = 1/9 \sum_{i=1}^{10} (\hat{p}_{Ki} - \hat{p}_K)^2$$

The results are shown in Table 3.5 which summarizes the variances obtained with and without systematic sampling. It can be seen that the results are rather mixed, and in some cases a worse result was obtained using systematic sampling. Most of this variation is strictly statistical fluctuation due to the unavoidably large variances in the efficiency estimates. That this is the case may be seen from the efficiencies which are less than 1.0. Theoretically, systematic sampling should always reduce variance and efficiencies should always be 1.0 or greater. However, if efficiencies are close to 1, it is easy to get estimates which are just below 1.0.

Some of the variation in the systematic sampling efficiencies may also represent a variation in how sensitive a parameter is to the submarine starting location. The probability of kill (eventually, after enough localization) should not be as dependent on the submarine's starting distance as the time taken to localize and kill or the number of sonobuoys used in localization. Any parameter which is not sensitive to submarine starting position will have a variance which is likewise not sensitive to reduction of variance in selecting the submarine starting position.

TABLE 3.5

Variance Reduction Using Systematic Sampling For The Long Range MAD Gear

| Estimated Expected Values | Variance With Crude Sampling | Variance With Systematic Sampling | Efficiency |
|---|---|---|---|
| Probability of Submarine Kill | $2.4 \times 10^{-3}$ | $2.5 \times 10^{-3}$ | 0.96 |
| Time to Submarine Kill | 9.6 | 9.0 | 1.07 |
| Number of Torpedos per Kill | $1.02 \times 10^{-3}$ | $5.75 \times 10^{-4}$ | 1.78 |
| Sonobuoys Used per Kill - Type A | $7.3 \times 10^{-2}$ | 0.102 | 0.71 |
| Sonobuoys Used per Kill - Type B | 0.16 | $7.3 \times 10^{-3}$ | 2.19 |

Average Efficiency Improvement $\cong$ 1.34
Ratio of time increase with variance reduction $\cong$ 1.00
Systematic Sampling: The initial submarine positions were distributed over ten strata using one random number per ten starting positions.

The average efficiency gain for systematic sampling in the long MAD problem was a factor of 1. 34.

A second type of systematic sampling was used in the problem with the short MAD gear. In this case, a new random number was used each time a new starting position for the submarine was selected. Ten strata were used with the starting position limited to fall between ranges $L_1$ and $L_2$. That is, the first ten starting positions were selected using

$$y_1 = L_1 + \frac{(L_2 - L_1)}{10} R_{u_1}$$

$$y_2 = L_1 + \frac{(L_2 - L_1)}{10} (R_{u_2} + 1.)$$

$$y_3 = L_1 + \frac{(L_2 - L_1)}{10} (R_{u_3} + 2.)$$

$$\vdots$$

$$y_{10} = L_1 + \frac{(L_2 - L_1)}{10} (R_{u_{10}} + 9.)$$

where $R_{u_1}, \cdots, R_{u_{10}}$ are random samples from U(0, 1).

The samples in this case were batched as before and the results presented in Table 3. 6 were obtained. These results are seen to be quite similar to those obtained for the long MAD gear. Again, a large variance in the efficiency is apparent. Also, it appears that several of the parameters were insensitive to initial submarine starting position. The average efficiency is 1. 28 with an overall uncertainty of about 20% expected in the efficiency estimate.

TABLE 3.6

Variance Reduction Using Systematic Sampling For The Short Range MAD Gear

| Estimated Expected Values | Variance With Crude Sampling | Variance With Systematic Sampling | Efficiency |
|---|---|---|---|
| Probability of Submarine Kill | $2.12 \times 10^{-3}$ | $2.3 \times 10^{-3}$ | 0.92 |
| Time to Submarine Kill | 19.2 | 16.8 | 1.08 |
| Number of Torpedos per Kill | $7.85 \times 10^{-4}$ | $9 \times 10^{-4}$ | 0.87 |
| Sonobuoys Used per Kill - Type A | 0.22 | 0.16 | 1.38 |
| Sonobuoys Used per Kill - Type B | 0.36 | 0.168 | 2.14 |

Average Efficiency Improvement $\cong 1.28$

Ratio of time increase with variance reduction $\cong 1.00$

Systematic Sampling: The initial submarine positions were distributed over ten strata using one random number for each starting position.

## 3.5 ANTITHETIC VARIATES FOR SAMPLING INITIAL SUBMARINE POSITION

The final variance reduction technique applied to selection of the submarine starting position was the use of antithetic variates. This was performed for the same two situations described for systematic sampling in the previous sections.

In its simplest application, use of antithetic variates seeks to generate negatively correlated samples by selecting two values $x'$, $x''$ of the random variable from the distribution $f(x)$ using

$$R_u = \int_{-\infty}^{x'} f(x)dx$$

and

$$1-R_u = \int_{-\infty}^{x''} f(x)dx$$

where $R_u$ is a random number selected from $U(0,1)$. The values of $x'$ and $x''$ are clearly correlated since they have been generated by the same random number $R_u$. Also, $x'$ and $x''$ are negatively correlated since when $x'$ is large, $x''$ will be small.

In the application here, pairs of initial starting positions for the submarine were selected according to the above formulation. Thus, when one submarine starting position was selected far from the aircraft, a position close to the aircraft was also selected for the next history.

In the first problem, (i.e., where the long range MAD gear was used and the submarine was located between 0 and L), the pairs of starting positions were obtained using

$$y_1 = R_u L$$

$$y_2 = (1 - R_u)L$$

and two histories were run.

In the second problem (i.e., with the short range MAD gear and where the submarine was located between $L_1$ and $L_2$) the pairs of starting positions were obtained using

$$y_1 = L_1 + (L_2 - L_1)R_u$$

$$y_2 = L_2 + (L_1 - L_2)R_u$$

and the two histories corresponding to these initial starting positions were run.

In both cases, batching was performed to obtain estimates for the variances. The results of the analyses are summarized in Tables 3.7 and 3.8 respectively.

As was the case with systematic sampling, there was a wide variation in efficiency and one variable gave worse results with the antithetic variates than with crude sampling, indicating as before, a large variance for the efficiency estimates and, possibly, several parameters which were not sensitive to the initial starting values. Although it is theoretically possible that the use of antithetic variates could give worse results than crude sampling, it was not expected here and the single value less than 1.0 is probably a low estimate for an efficiency just above 1.0. In any event, average efficiencies of 1.12 and 1.37 were obtained in the two problems. An error of about ±10% is expected in these efficiencies.

## 3.6 CORRELATED SAMPLING FOR ESTIMATING DIFFERENCES IN MAD GEAR EFFECTIVENESS

Correlated sampling is a procedure that can be used to reduce variance in Monte Carlo simulation in the following general situations:

- The effect of a perturbation to a known problem is to be determined.

TABLE 3.7

Variance Reduction Using Antithetic Variates For The Long Range MAD Gear

| Estimated Expected Values | Variance With Crude Sampling | Variance With Systematic Sampling | Efficiency |
|---|---|---|---|
| Probability of Submarine Kill | $2.4 \times 10^{-3}$ | $2.4 \times 10^{-3}$ | 1.00 |
| Time to Submarine Kill | 9.6 | 8.4 | 1.14 |
| Number of Torpedos per Kill | $1.02 \times 10^{-3}$ | $7.85 \times 10^{-4}$ | 1.31 |
| Sonobuoys Used per Kill - Type A | $7.3 \times 10^{-2}$ | $7.3 \times 10^{-2}$ | 1.00 |
| Sonobuoys Used per Kill - Type B | 0.16 | 0.137 | 1.17 |

Average Efficiency Improvement $\cong 1.12$
Ratio of time increase with variance reduction $\cong 1.00$
Antithetic Variates: The initial pairs of submarine starting conditions were obtained from $y_1 = LR_u$ and $y_2 = L(1-R_u)$.

TABLE 3.8

Variance Reduction Using Antithetic Variates For The Short Range MAD Gear

| Estimated Expected Values | Variance With Crude Sampling | Variance With Systematic Sampling | Efficiency |
|---|---|---|---|
| Probability of Submarine Kill | $2.12 \times 10^{-3}$ | $2.31 \times 10^{-3}$ | 0.92 |
| Time to Submarine Kill | 19.2 | 16 | 1.11 |
| Number of Torpedos per Kill | $7.85 \times 10^{-4}$ | $5.75 \times 10^{-4}$ | 1.36 |
| Sonobuoys Used per Kill - Type A | 0.22 | 0.168 | 1.32 |
| Sonobuoys Used per Kill - Type B | 0.36 | 0.168 | 2.14 |

Average Efficiency Improvement $\cong 1.37$

Ratio of time increase with variance reduction $\cong 1.00$

Antithetic Variates: Initial pairs of submarine starting conditions were obtained from $y_1 = L_1 + (L_2 - L_1)R_u$ and $y_2 = L_2 + (L_1 - L_2)R_u$

- The difference between estimated parameters in two problems having similar characteristics is to be calculated.

- A parametric study of several similar problems is to be performed.

Such situations occur very frequently. In fact, in most studies the most important result to be investigated is the change in response of the system as a problem characteristic is varied. As will be seen, the payoff from various types of correlation in sampling can be very high.

Use of the APAIR model could easily involve problems having one or more of these characteristics. For example, the sensitivity to a range of tactics presents a potential situation where correlated sampling could provide substantial improvements in efficiency.

The problem selected for demonstration of correlated sampling involved the short range and long range MAD detector cases discussed previously. The main parameter of interest to be calculated was the difference between these two cases in the parameters used in prior sections. That is, differences in:

- Probability of submarine kill

- Time to submarine kill

- Number of torpedos used per kill

- Number of sonobuoys of Types A and B used per kill.

The only difference in problem characteristics in the two cases was a difference in MAD detection capabilities. This was expressed in a function relating probability of detection to range of target from the aircraft, as shown in Fig. 3.2.

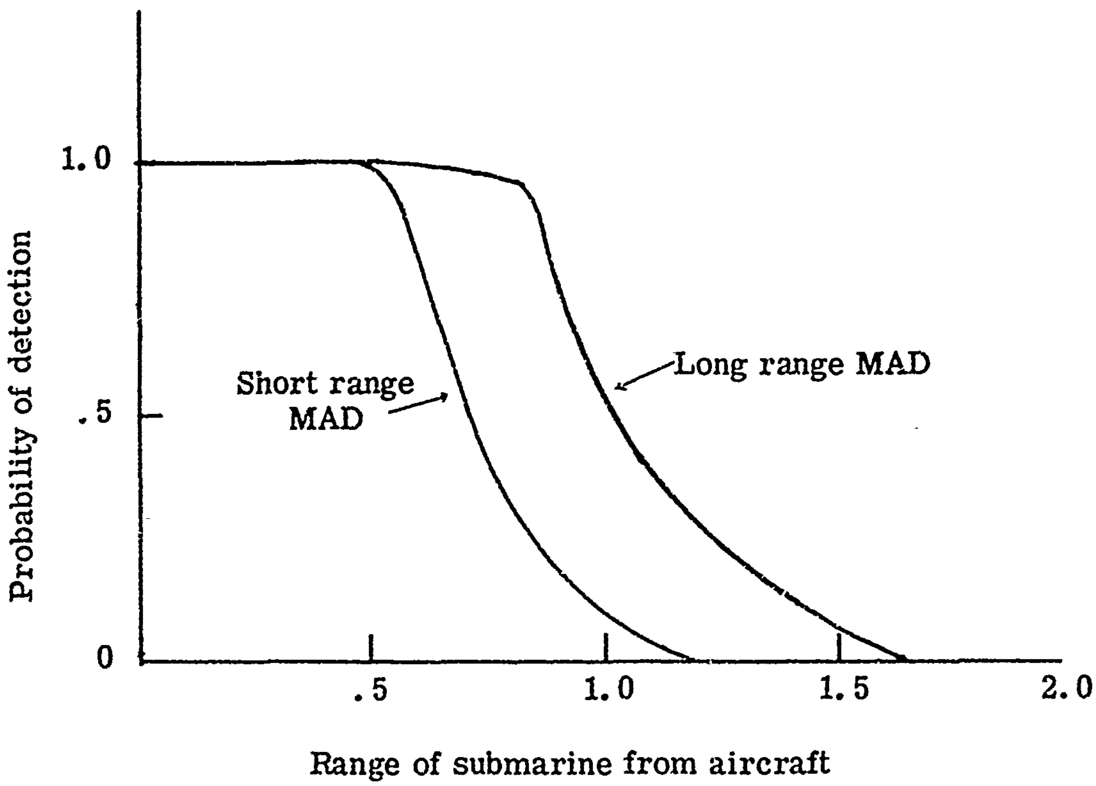



Fig. 3.2 Detection probabilities for the long range and short range MAD gear used in the APAIR studies

To demonstrate the concept of correlated sampling for this application, we define

$p_S$ = probability of submarine kill using the short range MAD detectors

$p_L$ = probability of submarine kill using the long range MAD detectors

The problem of interest is to perform Monte Carlo simulations to estimate the difference

$$\Delta = p_L - p_S$$

in the case of probability of submarine kill. Similar definitions apply to the remainder of the parameters of interest.

The crude Monte Carlo approach would first estimate $p_S$ (denoted by $\hat{p}_S'$) and then $p_L$ (denoted by $\hat{p}_L'$) using another set of independent histories. Then the difference $\Delta$ is estimated using

$$\hat{\Delta}' = \hat{p}_L' - \hat{p}_S'$$

The variance in $\hat{\Delta}'$ is

$$\sigma^2(\hat{\Delta}') \quad \sigma^2(\hat{p}_S') + \sigma^2(\hat{p}_S')$$

for the case of independent histories in the $p_S$ and $p_L$ estimations. Suppose, however, that positively correlated estimates for $p_S$ (say $\hat{p}_S$) and for $p_L$ (say $\hat{p}_L$) were used to estimate $\Delta$. Then for

$$\hat{\Delta} = \hat{p}_L - \hat{p}_S$$

the variance in $\hat{\Delta}$ will be given by

$$\sigma^2(\hat{\Delta}) = \sigma^2(\hat{p}_L) + \sigma^2(\hat{p}_S) - 2 \operatorname{cov}(\hat{p}_S, \hat{p}_L)$$

where $\operatorname{cov}(\hat{p}_S, \hat{p}_L)$ is the covariance between $\hat{p}_S$ and $\hat{p}_L$. Since $\hat{p}_S$ and

$\hat{p}_L$ are positively correlated, then

$$\text{cov}\,(\hat{p}_S, \hat{p}_L) \geq 0, \text{ and hence}$$

$$\sigma^2(\hat{\Delta}) \leq \sigma^2(\hat{\Delta}') \,.$$

Thus, the objective of correlated sampling is to develop a sampling strategy that will correlate the estimators $\hat{p}_S$ and $\hat{p}_L$. This technique can be extremely powerful when small perturbations of problems are to be studied since the correlations induced will tend to emphasize differences in the problems due to the perturbation rather than differences due to statistical fluctuations which are usually the controlling differences in cases with independent histories.

The above arguments for improvement in the variance using correlated sampling for the probability of kill would also apply directly to estimating the differences in other ASW parameters.

Correlation can be accomplished in several ways. For example, the short and long range MAD problems could be simulated independently except the same random number could be used in determining the outcome once the probability of detection had been obtained from the curves in Fig. 3.2. Thus, correlation between the two results would exist. Another way, and the one that was used here, is to control the random numbers in the two simulations, by using the same random numbers in the two problems until a difference in detection occurs in the problems due to the difference in the MAD gear. Two separate runs were made, but corresponding histories in the two simulations were made to start at the same point in the random number sequence. Thus, the histories would be identical up to the point where the difference in MAD gear resulted in different decisions. At the time the detectors came into play, the detection outcome was selected in each case from the same random number. Subsequently, the histories continued independently until the end of the game. More

correlation could have been introduced by subsequently using identical random numbers wherever the problem logic allowed.

The program changes made to induce this much correlation were fairly simple. Two separate random number generators were used in each simulation. The first was used once each history to give a random starting point in the sequence of the second generator; it produced the same sequence of starting points in both simulations. The second generator was used in the history to obtain random numbers for the simulation process. By starting at the same point in both cases, identical histories will be generated until there is a difference in decisions made concerning a MAD detection. Even though the first history in one problem might use more random numbers than in the other problem, the random number sequences would be returned to the same point at the start of the second history in each problem.

To calculate the variance of the difference, $\Delta$, in the two cases, it was necessary to obtain 'batch' values, $\hat{\Delta}_n$, for the difference in the batch values $\hat{p}_{Ln}$ and $\hat{p}_{Sn}$ that have been described in earlier sections. The batch values of $\hat{p}_{Sn}$, $\hat{p}_{Ln}$, and the other parameters were written on temporary files as the simulations progressed. Then a separate small program was written to combine these files and calculate the batch differences in the various parameters. The final estimated difference was the average of the batch differences and the variance of the difference was determined from the spread of the sample batch differences. Specifically, by grouping the 100 histories of the long MAD simulation into batches of 10, one calculates

$$\hat{p}_{L1} = 1/10 \sum_{i=1}^{10} p_{Ki}$$

$$\hat{p}_{L2} = 1/10 \sum_{i=11}^{20} p_{Ki}$$

$$\vdots$$

$$\hat{p}_{L10} = 1/10 \sum_{i=91}^{100} p_{Ki}$$

where $\hat{p}_{Ln}$ is the estimate of $p_K$ from batch $n$ and $p_{Ki}$ is the estimate of $p_K$ from history $i$. Similar formulas were used to obtain batch estimates for the other parameters in the long and short simulations. Then the batch differences were calculated:

$$\hat{\Delta}_1 = \hat{p}_{L1} - \hat{p}_{S1}$$

$$\vdots$$

$$\hat{\Delta}_{10} = \hat{p}_{L10} - \hat{p}_{S10} \ .$$

The final estimator for the difference in probability of kill is

$$\hat{\Delta} = 1/10 \sum_{n=1}^{10} \hat{\Delta}_n$$

and the estimated variance is

$$\hat{\sigma}^2(\hat{\Delta}) = \frac{1}{10}\left[\frac{1}{9}\sum_{n=1}^{10} (\hat{\Delta}_n - \hat{\Delta})^2\right] \equiv \frac{1}{9}\left[\frac{1}{10}\sum_{n=1}^{10} \hat{\Delta}_n^2 - \left(1/10 \sum_{n=1}^{10} \hat{\Delta}_n\right)^2\right]$$

Rather than make two additional uncorrelated runs to get comparison variance estimates for the uncorrelated or crude Monte Carlo, the uncorrelated equation

$$\hat{\sigma}^2(\hat{\Delta}) = \hat{\sigma}^2(\hat{p}_S) + \hat{\sigma}^2(\hat{p}_L)$$

was used with

$$\hat{\sigma}^2(\hat{p}_S) = \frac{1}{9}\left[\frac{1}{10}\sum_{n=1}^{10} \hat{p}_{Sn}^2 - \left(1/10 \sum_{n=1}^{10} \hat{p}_{Sn}\right)^2\right]$$

and

$$\hat{\sigma}^2(\hat{p}_L) = \frac{1}{9}\left[\frac{1}{10}\sum_{n=1}^{10} \hat{p}_{Ln}^2 - \left(1/10 \sum_{n=1}^{10} \hat{p}_{Ln}\right)^2\right]$$

The results of the calculations are shown in Table 3.9. The greatest increase in efficiency (a factor of 3) was found for the probability of kill. The average improvement over all parameters was almost a factor of 2. Thus, the running time for the same variance could be reduced by about a factor of 2 by using correlated sampling.

There was a slight increase in running time (although the variance was substantially reduced) with variance reduction since some additional bookkeeping was used in the program. The running time could have been reduced with some additional effort by not recalculating with identical random numbers the part of each history up to the point where the detection came into play, but by simply saving the results of one case for application to the other. However, this would have involved more extensive computer program modifications than were warranted here.

Analysis of Correlated Histories

One of the great benefits of correlation is the potential it provides to gain insight into understanding simulation problems. For example, if two highly correlated histories, one with and one without a problem perturbation were available, then, most of the time, differences observed in the histories will be due to variations in the problem perturbation rather than statistical variations.

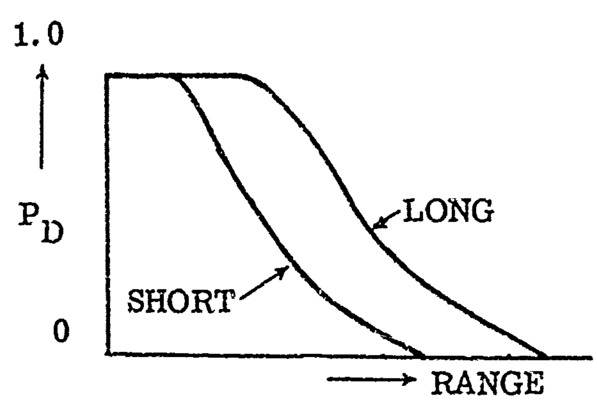
The problem of the two MAD detectors was ideally suited for demonstrating the possibility of analysis of correlated histories since the MAD detectors

TABLE 3.9

Variance Reduction Using Correlated Sampling For Estimating Differences Between Short and Long Range MAD Gear Parameters

| Estimated Difference in Expected Value (Long Range MAD - Short Range MAD) | Variance With Crude Sampling | Variance With Correlated Sampling | Efficiency |
|---|---|---|---|
| Probability of Submarine Kill | $2.96 \times 10^{-3}$ | $9.6 \times 10^{-4}$ | 3.0 |
| Time to Submarine Kill | 15.0 | 9.2 | 1.58 |
| Number of Torpedos per Kill | $2.2 \times 10^{-3}$ | $1.6 \times 10^{-3}$ | 1.33 |
| Sonobuoys Used per Kill - Type A | 0.166 | 0.068 | 2.40 |
| Sonobuoys Used per Kill - Type B | 0.174 | 0.144 | 1.17 |

Average Efficiency Improvement ≅ 1.9
Ratio of time increase with variance reduction ≅ 1.03
Correlated Sampling: Two separate runs were made. Histories were identical up to point where difference in MAD gear lead to a divergence in histories.



were sufficiently similar so that variations on estimates for the kill probability could be lost in the statistics of the problem. More specifically, consider the sample series of single histories shown in Table 3.10. These histories were obtained from runs correlated in the manner described above.

There are three types of situations that can be identified in Table 3.10. First, there are a large number of histories where the short and long range gear give the same result. This situation should be expected since the gear is similar and the histories are highly correlated. That is, when a short range detection occurs with the short MAD, it will also occur with the long MAD. Also when no detection occurs for the long MAD at long ranges, none will occur for the short MAD.

At intermediate ranges where the two MAD curves differ, this, of course, is not true. This is manifested by the second situation (history 8) where the long range MAD detected the submarine and effected a kill and the short range MAD didn't, with no kill as a result. This type of history was also expected.

Of most interest is the third situation where, in histories 1 and 9, the long range MAD gear produces a lower kill probability than the short MAD gear although both types detected the submarine. This result was quite unexpected and would tend to indicate there are considerations other than variations in MAD gear which make the two problems different. For example, the tactics used might be appropriate to the short MAD detector but might be 'trigger happy' when used with the long MAD gear resulting in premature torpedo drops with a lower kill probability. Had the histories not been correlated, it would have been difficult, due to the statistical variation from history to history, to notice that such events were occurring. In this manner, therefore, correlation can identify which histories should be examined in detail to provide more insight into the problem.

TABLE 3.10

Analysis of Correlated Histories for Long Range and Short Range MAD Simulations

| History No. | $P_K$ (Short MAD) | $P_K$ (Long MAD) | |
|---|---|---|---|
| 1 | .96 | .80 | (3) |
| 2 | .80 | .80 | (1) |
| 3 | .00 | .00 | (1) |
| 4 | .80 | .80 | (1) |
| 5 | .80 | .80 | (1) |
| 6 | .00 | .00 | (1) |
| 7 | .80 | .80 | (1) |
| 8 | .00 | .96 | (2) |
| 9 | .80 | .68 | (3) |
| 10 | .80 | .80 | (1) |

(1) In these histories the difference in MAD gear had no effect. A large number of such occurrences is reasonable due to correlation.

(2) In this history, the long range MAD gear picked up the submarine, resulting in a kill while the short range gear missed detection. This type of result is expected.

(3) In these histories, detection was accomplished with both types of gear. However, the kill probability is lower with the better gear which is unexpected. Correlation thus pinpoints which histories should be studied in detail to gain further insight.

Another application of correlated sampling is presented in the following section.

## 3.7 HISTORY REANALYSIS FOR ESTIMATING DIFFERENCES IN MAD GEAR EFFECTIVENESS

In the second application of correlated sampling to APAIR, reanalysis of histories using weight factors to correct for the difference in problem characteristics was used rather than controlling the random numbers. Effectively the following was performed for history j:

- A base history was run using crude sampling for the long MAD gear up to the point where the possibility of detection was to be tested.
- Given the range, a probability of detection for the long range gear ($p_L$) and the short range gear ($p_S$) were determined from the curves shown in Fig. 3.3.
- A random number was compared to $p_L$ to determine if a detection occurred and the history continued, using the results of that random decision.
- Weighting factors were assigned to the history according to the following rules:

  - If the first test for a detection with the long gear resulted in a detection, a weight correction factor of $W_{1j} = p_S/p_L$ was assigned to account for the short range gear. If the first attempt resulted in no detection, he the assigned weighting factor was

$$W_{1j} = \frac{1-p_S}{1-p_L} .$$

  - The history continues for subsequent tests of detection with the long MAD gear by assigning weights according to

$$W_{i+1,j} = \frac{p_S}{p_L} \cdot W_{i,j}$$

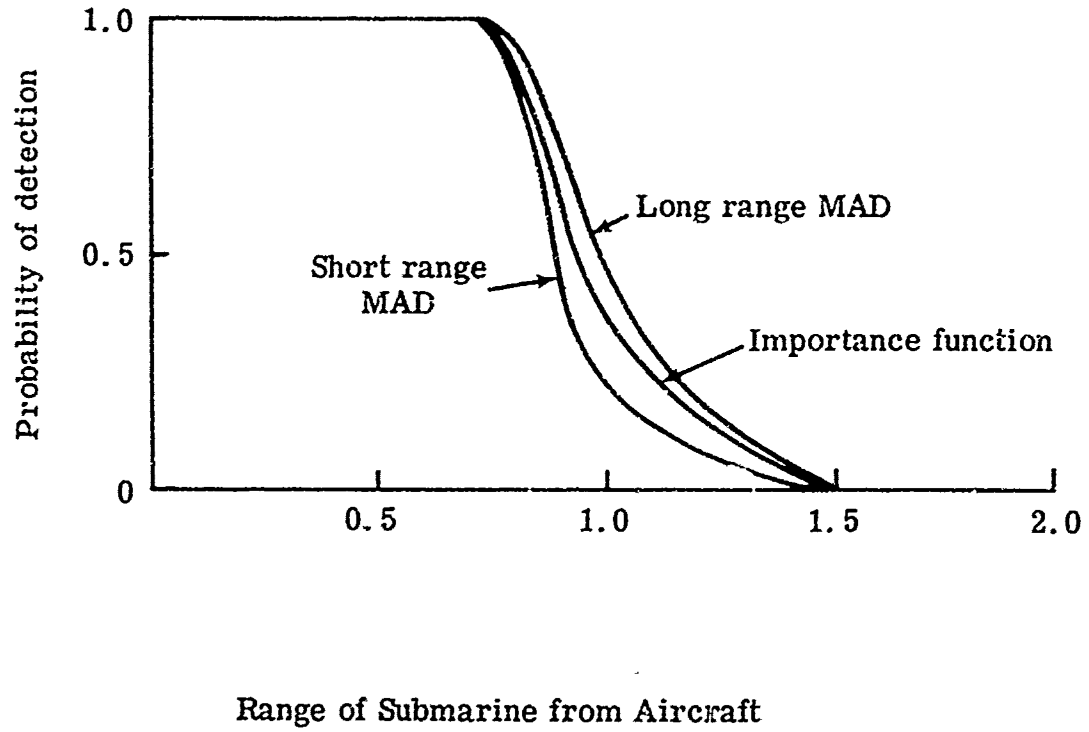



Fig. 3.3 Probability of detection versus range for the MAD detectors used in the demonstration of history reanalysis and importance sampling.

if a detection occurred on the i+1 st test and

$$W_{i+1,j} = \frac{1-p_S}{1-p_L} \cdot W_{i,j}$$

when a nondetection occurred on the i+1 st use of the MAD gear. This is continued until the game stops due to a kill or some other reason (e.g., sonobuoy stores exhausted). The final weighting factor is defined as $W_j$.

- If a kill occurred in this history then

$$n_j = 1$$

otherwise

$$n_j = 0$$

(Note that the total number of kills in N histories is simply

$$N_k = \sum_{j=1}^{N} n_j \; )$$

The above series of steps was performed for the N histories and the following formulas were used to estimate the differences in the parameters of interest.

Consider first the probability of kill. The estimated probability of kill for the long gear is given by

$$\hat{p}_L = 1/N \sum_{j=1}^{N} n_j$$

The estimated probability of kill for the short gear is given by

$$\hat{p}_S = 1/N \sum_{j=1}^{N} n_j W_j$$

The difference in probability of kill (long MAD - short MAD) is

$$\hat{\Delta} = 1/N \sum_{j=1}^{N} n_j (1-W_j)$$

Similar considerations were used for the time to kill. If history j resulted in a kill and the time of kill was $T_j$, the average time to kill for the long MAD is

$$\hat{T}_L = 1/N_k \sum_{j=1}^{N} n_j T_j$$

while for the short MAD it is

$$\hat{T}_S = \frac{1}{\hat{p}_S N} \sum_{j=1}^{N} n_j T_j W_j \quad .$$

($\hat{p}_S N$ is the 'number' of kills in the short MAD problem and is the correct normalizing factor for the time to kill and other parameters.) In a similar manner the remaining parameters (number of torpedos or sonobuoys used per kill) were calculated.

It is clear the results obtained for long and short range detectors are highly correlated. The histories for the two cases are not only identical up to the point of possible detection, but they continue to be correlated throughout. When a kill is registered due to the use of the long range gear, it is also registered for the short range gear but carries a different weight. As the random choices made are appropriate to the long gear, they will not be

optimum for the short range gear. This will increase the variance of the short range gear estimates, but the high degree of correlation should still reduce the total variance of the difference.

A second significant advantage of the above approach is that the results of two problems have been obtained by actually performing only one simulation (i.e., for the long gear) although some additional computation is required to perform the weighting. However, the total computing time required was only 53% of that needed to perform two independent calculations.

Thus, to generate the required correlated cases, approximately one-half of the computational effort was required. In addition to this time saving, there were substantial improvements in the variances of the different parameters estimated. These results are summarized in Table 3.11 where it is seen that substantial improvements in the efficiency were realized. For example, in estimating the stores used (torpedos and Type B buoys) about a factor of 10 improvement in efficiency was obtained. The overall average efficiency was found to be almost 7. This can, of course, be interpreted as a reduction in computational time that can be realized using the correlated sampling scheme described here.

The short range MAD detection probability curve used in this technique and that described in the following section is shown in Fig. 3.3. This differs from the short range gear shown in Fig. 3.2 which was used in the other studies of this report. This change was made because use of the curve in Fig. 3.2 would have resulted in a considerable number of identically zero weights whenever a detection occurred beyond the range of the short gear. This would have destroyed much of the correlation in the histories and counteracted the effect we were trying to illustrate. Therefore, a dummy short MAD gear curve having the same limits of range as the long range gear but with a lowered probability of detection at the longer ranges was used. Paradoxically, this had the effect of making the two types of detector more similar, thus making it harder to calculate the difference between them.

TABLE 3.11

Variance Reduction Using History Reanalysis For Estimating Differences Between Short and Long Range MAD Gear

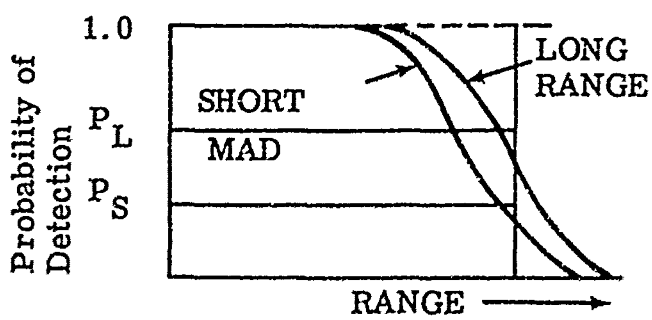
| Estimated Difference in Expected Value (Long Range - Short Range) | Variance With Crude Sampling | Variance With History Reanalysis | Efficiency |
|---|---|---|---|
| Probability of Submarine Kill | $4.2 \times 10^{-3}$ | $6.6 \times 10^{-3}$ | 1.21 |
| Time to Submarine Kill | 7.8 | 2.05 | 7.2 |
| Number of Torpedos per Kill | $1.25 \times 10^{-2}$ | $2.56 \times 10^{-3}$ | 9.3 |
| Sonobuoys Used per Kill - Type A | 0.097 | 0.0289 | 6.4 |
| Sonobuoys Used per Kill - Type B | 0.204 | 0.040 | 9.7 |

Average Efficiency Improvement $\simeq 6.8$

Ratio of time decrease with variance reduction $\simeq 0.53$

History Reanalysis: One run was made using the long range gear probability distribution for detection. Simultaneously, calculations were made for short range gear using weighting factors to correct for difference in probability between the two distributions.

| | Detection ($P_L$) | No Detection ($1-P_L$) |
|---|---|---|
| Weight correction for Short Gear | $W \cdot \frac{P_S}{P_L}$ | $W \cdot \frac{1-P_S}{1-P_L}$ |



The estimated differences in the various parameters and their variances were calculated by the same batching techniques as described in the previous section. For this technique the uncorrelated difference variances were determined by

$$\hat{\sigma}^2(\hat{\Delta}) = \hat{\sigma}^2(\hat{p}_L) + \hat{\sigma}^2(\hat{p}_S) \approx 2\hat{\sigma}^2(\hat{p}_L)$$

where $\hat{\sigma}^2(\hat{p}_L)$ was determined from the batch values of $\hat{p}_L$. Due to the weight corrections, a $\hat{\sigma}^2(\hat{p}_S)$ calculated from the reanalyzed histories would have been much larger than $\sigma^2(\hat{p}_S)$ from an uncorrelated case. Rather than make a separate uncorrelated run, it was simply assumed that $\hat{\sigma}^2(\hat{p}_S) \simeq \sigma^2(\hat{p}_L)$ .

A third variation involving correlated sampling will be described next.

## 3.8 IMPORTANCE SAMPLING WITH CORRELATION ESTIMATING DIFFERENCES IN MAD GEAR EFFECTIVENESS

To illustrate the technique of importance sampling, which has wide applicability at many stages of a Monte Carlo simulation, a demonstration involving an extension of the previous correlation problem was devised. It was very similar to the calculation of the previous section except that, in place of the long range MAD detection probability, an 'importance function' detection probability, mid-way between the long and short range curves as shown in Fig. 3.3, was used in making detect/no detect decisions in the game.

As explained above, the use of the long MAD curve to generate the histories results in choices which are not appropriate to the short MAD gear. This increases the variance of the short MAD part of the calculation. The use of an 'importance function' curve generates choices which are more optimum for the short gear but less optimum for the long MAD. It was hoped that the result would be a reduced variance overall.

The procedure used in this simulation was a mere extension of the procedure described in the previous section. That is, for history j the following sequence of steps took place:

- A base history was run using crude sampling. At points in the problem where detection was to be tested, the importance function was used in a random determination of whether or not a detection occurred.

- From the range, detection probabilities were derived from the curves for the 'importance' ($p_I$), the long MAD ($p_L$), and the short MAD ($p_S$) as given in Fig. 3.3.

- A random number was compared to $p_I$ to determine if detection occurred.

- If this first test resulted in a detection, an assigned weight correction for the short range gear was set to

$$W_{1j}^{S} = \frac{p_S}{p_I}$$

and a weight for the long range gear of

$$W_{1j}^{L} = \frac{p_L}{p_I}$$

was assigned.

If no detection occurred, the respective weights assigned were

$$W_{1j}^{S} = \frac{1-p_S}{1-p_I}$$

and

$$W_{1j}^{L} = \frac{1-p_L}{1-p_I}$$

- The history continued for subsequent tests of detection by assigning respective weights according to:

$$W^S_{i+1,j} = \frac{p_S}{p_I} W^S_{i,j}$$

for detection on the i+1st test for submarine detection and

$$W^S_{i+1,j} = \frac{1-p_S}{1-p_I} W^S_{i,j}$$

for nondetections on the $i\pm1$ st test with similar equations for $W^L$. This is continued until a kill occurs or the history is otherwise terminated. The final weighting factors calculated in the history are defined as $W^S_j$ and $W^L_j$.

- If a kill occurred in this history, then

$$n_j = 1$$

otherwise

$$n_j = 0$$

(the total number of kills in N histories is

$$N_k = \sum_{j=1}^{N} n_j).$$

The above series of steps was performed for the N histories and the following formulas were used to estimate the differences in the parameters of interest.

For the probability of kill, the estimate for the short range gear was given by

$$\hat{p}_S = 1/N \sum_{j=1}^{N} n_j W^S_j$$

and for the long range gear,

$$\hat{p}_L = 1/N \sum_{j=1}^{N} n_j W_j^L$$

The difference is simply

$$\hat{\Delta} = \hat{p}_L - \hat{p}_S = 1/N \sum_{j=1}^{N} n_j (W_j^L - W_j^S)$$

Similarly, if $T_j$ is the time to kill on history $j$ (assuming a kill occurred), the average time to kill a submarine would be given by

$$\hat{T}_L = \frac{1}{\hat{p}_L N} \sum_{j=1}^{N} n_j T_j W_j^L$$

and

$$\hat{T}_S = \frac{1}{\hat{p}_S N} \sum_{j=1}^{N} n_j T_j W_j^S$$

The number of torpedos and sonobuoys used were calculated in a similar manner.

The correlation between the results for the long and short MAD detectors in this case arises from the fact that their estimates are derived from the same importance sampled set of histories. Also, it is important to recognize that the running time is approximately one-half of that required to run two independent cases. In fact, it was found that the ratio of the running time with and without the correlation was a factor of 0.53, which is the same as the result in the previous case.

As expected, there was substantial improvement in the variance of the estimates. The results are summarized in Table 3.12. It can be seen that a factor of almost 20 was achieved in the efficiency of the estimator for the difference in the number of torpedos used. Also, a substantial improvement (a factor of 8.5) in the variance of the difference in the kill probability was realized. An average efficiency of 7.2 was found which, as before, can be construed as a direct factor for improvement in problem running time.

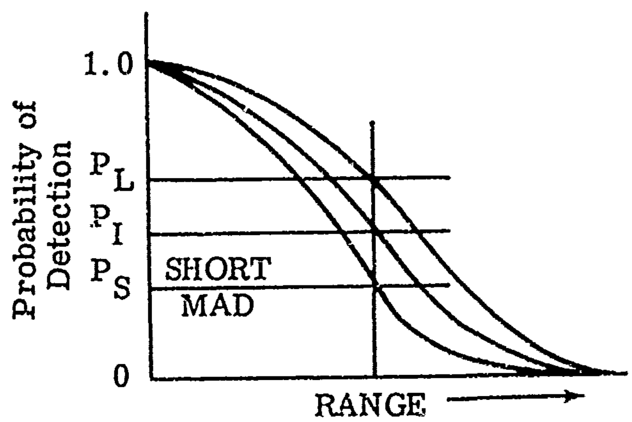
TABLE 3.12

Variance Reduction Using Importance Sampling For Estimating Differences Between Short and Long Range MAD Gear

| Estimated Differences in Expected Values (Long Range - Short Range) | Variance With Straightforward Sampling | Variance With Correlation | Efficiency |
|---|---|---|---|
| Probability of Submarine Kill | $4.2 \times 10^{-3}$ | $1.22 \times 10^{-3}$ | 8.5 |
| Time to Submarine Kill | 7.8 | 8.3 | 1.8 |
| Number of Torpedos per Kill | $1.25 \times 10^{-2}$ | $1.21 \times 10^{-3}$ | 19.7 |
| Sonobuoys Used per Kill - Type A | 0.097 | 0.062 | 3.0 |
| Sonobuoys Used per Kill - Type B | 0.204 | 0.077 | 5.0 |

Average efficiency improvement $\cong 7.2$
Ratio of time decrease with variance reduction $\cong 0.53$
Importance Sampling: One run was made using the importance function for selection. Simultaneously calculations were made for short and long range gear using weighting factors to correct for differences in the probability distributions.

| | Detection ($P_I$) | No Detection ($1-P_I$) |
|---|---|---|
| Weight correction (short gear) | $W^S \cdot \frac{P_S}{P_I}$ | $W^S \cdot \frac{1-P_S}{1-P_I}$ |
| Weight correction (long gear) | $W^L \cdot \frac{P_L}{P_I}$ | $W^L \cdot \frac{1-P_L}{1-P_I}$ |



# APPENDIX A

# MIRAN

# A MACHINE INDEPENDENT PACKAGE FOR GENERATING UNIFORM RANDOM NUMBERS

Preceding page blank

## APPENDIX A

## MIRAN - A MACHINE INDEPENDENT PACKAGE FOR GENERATING UNIFORM RANDOM NUMBERS

### A.1 GENERAL DISCUSSION

The standard technique for producing uniform random numbers on modern high-speed computers is an algorithm known as the multiplicative congruential method. This method is expressed mathematically as

$$R_{n+1} = \lambda \cdot R_n \text{ (modulo P)} .$$

Since the R's are integers ranging from 1 to P-1, successive real random numbers uniformly distributed from 0 to 1 are generated by dividing $R_n$ by P. The properties of this technique as a random number generator (RNG) are highly dependent on the choice of the generator, $\lambda$, and the modulus, P. Unfortunately, there are many RNGs in current use which do not approximate randomness closely enough to be sufficient for all Monte Carlo calculations and, what is far worse, do manage to pass some of the simple tests for randomness. There are, however, several choices of $\lambda$ and P which have been thoroughly tested, both theoretically[9] and through many years of actual use in Monte Carlo calculations, and which appear to be sufficiently random for general usage.

For reasons of convenience and efficiency, P is generally taken to be $2^m$ where m is the number of bits, excluding the sign bit, in a single word on the particular computer being used. The generation process starts with a fixed generator, $\lambda$, and a starting value, $R_0$. The full product from the multiplication of $\lambda$ and $R_0$ would usually fill two computer words; however, the modulo P in the algorithm means that we only need the single word, $R_1$, comprising the low order half of the $\lambda \cdot R_0$ product. The random number generation is completed by converting $R_1$ to a real variable and

dividing by P. $R_1$ replaces $R_0$ in storage in the random number subroutine and the process is ready to begin anew.

In this sort of a process there have been two barriers to developing a Fortran RNG subroutine which would be independent of the particular computer for which it was designed. The first is the modulus P, which varies from computer to computer as the word length varies. [Choosing a universal value of P to fit the smallest computer is not a good solution as the properties of a RNG become less random as P is made smaller, to the extent that Coveyou and MacPherson[9] consider them questionable for $P = 2^{31}$ (IBM 360 series) and borderline for $P = 2^{35}$ (IBM 7090, Univac 1108, etc.).] The second problem is that the sign bit of $R_1$ may need to be cleared following the multiplication. Clearing the sign bit generally requires some trickery in Fortran which varies from computer to computer as the mode of representation (one's complement, two's complement, uncomplemented, etc.) of negative numbers varies.

The way around these obstacles is to use an explicit multiple precision representation. The integers and operations involved in the RNG algorithm are separated into component parts in such a way that all operations are kept within a single computer word and no overflows into the sign bit are made, thus avoiding the sign-clearing problem. Through multiple precision a sufficiently large modulus for good RNG properties may be used even though the actual computer word size is small. An initialization call must be made to convey to the RNG the maximum integer allowed on the particular computer being used so that it can set up an appropriate multiple precision representation.

The advantage of a RNG that is machine independent is simple: it greatly facilitates the exchange and checkout of Monte Carlo programs between different computers. The price paid for this advantage is also simple: it is a much slower method of producing random numbers. However, it is

still fast enough (several thousand random numbers generated in one second) that the time difference will not be noticed in most Monte Carlo applications.

## A.2 CHOICE OF A SPECIFIC ALGORITHM FOR MIRAN

The work of Coveyou and MacPherson[9] has provided a thorough theoretical analysis of many commonly used RNGs. They show that the correlation properties of a RNG are strongly dependent on the modulus P. For values of $P = 2^{31}$ or $2^{35}$, there must necessarily be a waviness or graininess to the joint distribution of two, three, and four consecutive random numbers that could lead to incorrect results for some Monte Carlo calculations. For $P = 2^{47}$, the departures from true randomness are small enough as to be negligible for practical calculations. Among the specific generators, $\lambda$, tested by Coveyou and MacPherson, there is one, $\lambda = 5^{15}$, which has good statistical properties and which may be easily produced by a machine independent subroutine. (In a subroutine designed for use on computers of varying word length, specifying a fixed 47-bit integer through data statements would be difficult. However, $5^{15}$ may easily be produced by multiplying 5's after the exact multiple precision representation needed has been established.) In addition the choice of $P = 2^{47}$ and $\lambda = 5^{15}$ has an added advantage: this particular choice of a RNG has seen long usage (several thousand hours on a CDC 1604 at Oak Ridge National Laboratory) in Monte Carlo computations without any apparent problems.

## A.3 MULTIPLE PRECISION REPRESENTATION

In the basic algorithm used by MIRAN, $\lambda$ and the $R_n$ values will be 47-bit integers. This may exceed machine capacity. To keep all arithmetic operations from overflowing a single machine word, these integers are stored in an array wherein each word of the array constitutes a 'digit' in a representation of the integer to a particular base. This basis, called BASE, is chosen at execution time so that $(BASE)^2$ does not exceed the maximum integer allowed on the particular computer being used. Thus, for

example, on a machine with 35-bit words (unsigned), BASE would be $2^{17}$ and each 47-bit integer would be broken down into 3 words as follows:

| 47-bit Integer | Multiple Precision Representation | |
|---|---|---|
| $b_1 b_2 \ldots b_{13} b_{14} \ldots b_{30} b_{31} \ldots b_{47}$ | $+\,0 \ldots 0\; b_1 \ldots b_{13}$ | word 3 |
| | $+\,0 \ldots 0\; b_{14} \ldots b_{30}$ | word 2 |
| | $+\,0 \ldots 0\; b_{31} \ldots b_{47}$ | word 1 |

Note that the 'digits' are stored in the array in 'reverse' order, i.e., word 1 is the least significant 17 bits of the number. Also, since 17 does not go evenly into 47, the last word contains only 13 bits.

Arithmetic in a multiple precision representation is carried out in the same manner as arithmetic is normally done by hand. The addition of two numbers, for example, is done digit by digit. When two 'digits', or words, are added there may be an overflow into the $18^{th}$ bit of the result. This must be detected, the overflow cleared out, and a carry of 1 added into the next higher 'digit'. Multiplication is slightly more complex. It is again carried out digit by digit and the resulting products are added, keeping them in appropriate columns, to get the final product. The multiplication of two 'digits' produces, of course, a two-digit product which is initially contained in a single computer word. This must be broken down into a high-order digit and a low-order digit with the high-order digit being added into the next higher column of the result. As each column is added, a carry over into the next higher column may be needed. Thus, in our example where three words were used for each integer, nine multiplies and several additions would be needed to form the six-word full product as schematized below.

$$
\begin{array}{cccccc}
 & & & d_3 & d_2 & d_1 \\
 & & & d'_3 & d'_2 & d'_1 \\
\cline{4-6}
 & & & & h_{11} & \ell_{11} \\
 & & & h_{21} & \ell_{21} & \\
 & & h_{31} & \ell_{31} & & \\
 & & & h_{12} & \ell_{12} & \\
 & & h_{22} & \ell_{22} & & \\
 & h_{32} & \ell_{32} & & & \\
 & & h_{13} & \ell_{13} & & \\
 & h_{23} & \ell_{23} & & & \\
h_{33} & \ell_{33} & & & & \\
\hline
S_6 & S_5 & S_4 & S_3 & S_2 & S_1
\end{array}
$$

where $h_{ij}$ and $\ell_{ij}$ are the high and low order parts of the product of $d_i$ and $d'_j$.

## A.4 USE OF MIRAN PACKAGE

Initialization:

Before generating any random numbers, it is necessary to make an initialization call. This is done by the statement

```
CALL RANSET (MAXINT,NSTART)
```

where MAXINT is the maximum integer allowed on the computer (or compiler) being used. NSTART is the starting value, $R_0$, to be used in the random number sequence. If NSTART is less than or equal to 0, a default value of 2001 is supplied for NSTART. If NSTART is even, the next higher odd number will be used.

For example MAXINT = $2^{35}-1$ on a 1108, $2^{48}-1$ on a CDC-6600, etc. Good values for NSTART are any odd integer although frequent use of small odd integers is not recommended for calculations employing a relatively small number of random numbers.

The random numbers are generated in subroutine URAND which may be used as either a function subroutine or as an ordinary subroutine returning a value. Thus, either

```
CALL URAND(R)
        or
R  =  URAND(X)
```

will store a uniform random number in R. (Note that in the second form the same random number will also be stored in X. Thus, X must be a Fortran variable and not a constant.)

Limitations of MIRAN:

MIRAN will work on all computers where MAXINT is greater than 1023 and less than $2^{94}$. (These limits are practical and not theoretical and could be extended if it were ever necessary.)

## A.5 MIRAN PROGRAM DETAILS

The Fortran listings of the two MIRAN routines URAND and RANSET are presented in Figures A-1 and A-2. The accompanying logic flow is detailed in Figures A-3 and A-4. Additional explanation of the last step in the URAND logic is provided below.

The two subroutines URAND and RANSET communicate through a labelled common, MIRNG which contains

RAN(10) - An array containing the 'digits' of the current (or last) multiple precision random integer

```
      REAL FUNCTION URAND(FRAN)
      COMMON /[illegible]/ RAN(10),GEN(10),NWRD,BASE,MOD,FBASE,FMOD
      DIMENSION SUM(10)
      INTEGER RAN,GEN,BASE,CARRY,SUM,PROD,HPROD
      DO 30 IS=1,NWRD
   30 SUM(IS)=0
      DO 1 IR=1,NWRD
      N2=NWRD-IR+1
      DO 1 IG=1,N2
      IS=IR+IG-1
      PROD=RAN(IR)*GEN(IG)
      HPROD=PROD/BASE
      LPROD=PROD-HPROD*BASE
      SUM(IS)=SUM(IS)+LPROD
      IF (IS.LT.NWRD) SUM(IS+1)=SUM(IS+1)+HPROD
    1 CONTINUE
      N2=NWRD-1
      DO 5 IS=1,N2
      CARRY=SUM(IS)/BASE
      SUM(IS)=SUM(IS)-CARRY*BASE
      SUM(IS+1)=SUM(IS+1)+CARRY
    5 CONTINUE
      SUM(NWRD)=SUM(NWRD)-MOD*(SUM(NWRD)/MOD)
      DO 20 IS=1,NWRD
   20 RAN(IS)=SUM(IS)
      FRAN=SUM(1)
      DO 10 IS=2,NWRD
   10 FRAN=FRAN/FBASE+SUM(IS)
      FRAN=FRAN/FMOD
      URAND=FRAN
      RETURN
      END
```



Figure A-1. Fortran listing of URAND

```
      SUBROUTINE RANSET(MAXINT,NSTRT)
      COMMON /MIC46/ RAN(10),GEN(10),NWRD,BASE,MOD,FBASE,FMOD
      INTEGER RAN,GEN,BASE,CARRY,REM
      MAXI=MAXINT/4
      IB=0
      BASE=1
   99 IF (BASE.GT.MAXI) GO TO 100
      BASE=BASE*2
      IB=IB+1
      GO TO 99
  100 BASE=2**IB
      FBASE=BASE
      NWRD=47/IB+1
      REM=47-IB*(NWRD-1)
      MOD=2**REM
      FMOD=MOD
      DO 101 N=1,10
      RAN(N)=0
  101 GEN(N)=0
      GEN(1)=5
      DO 200 I=1,14
      CARRY=0
      DO 190 N=1,NWRD
      GEN(N)=GEN(N)*5+CARRY
      CARRY=0
      IF (GEN(N).LT.BASE) GO TO 190
      CARRY=GEN(N)/BASE
      GEN(N)=GEN(N)-BASE*CARRY
  190 CONTINUE
  200 CONTINUE
      NSTART=NSTRT
      IF (NSTART.LE.0) NSTART=2001
      NSTART=2*(NSTART/2)+1
      DO 300 N=1,NWRD
      NTEMP=NSTART/BASE
      RAN(N)=NSTART-NTEMP*BASE
  300 NSTART=NTEMP
      RETURN
      END
```



Figure A-2. Fortran listing of RANSET

START

30 Clear out SUM array

1 For $i = 1$, NWRD and $j = 1$, NWRD+1 - i:

Multiply $i^{th}$ 'digit' of RAN by $j^{th}$ 'digit' of GEN

Separate the 'two-digit' product into a high-order part HPROD and low-order part, LPROD

Add LPROD into the $(i + j-1)^{th}$ column of SUM

Add HPROD into the $(i + j)^{th}$ column of SUM

5 For $i = 1$, NWRD-1

Separate $i^{th}$ word of SUM into a single 'digit' plus the carry into the next higher column

Add the carry into the $(i + 1)^{th}$ word of SUM

Reduce the last word of SUM modulo MOD

Store SUM in RAN for next entry to URAND

Convert SUM to single precision floating point and divide it by P. Return this as the random number

END

Figure A-3. Logic flow chart for URAND

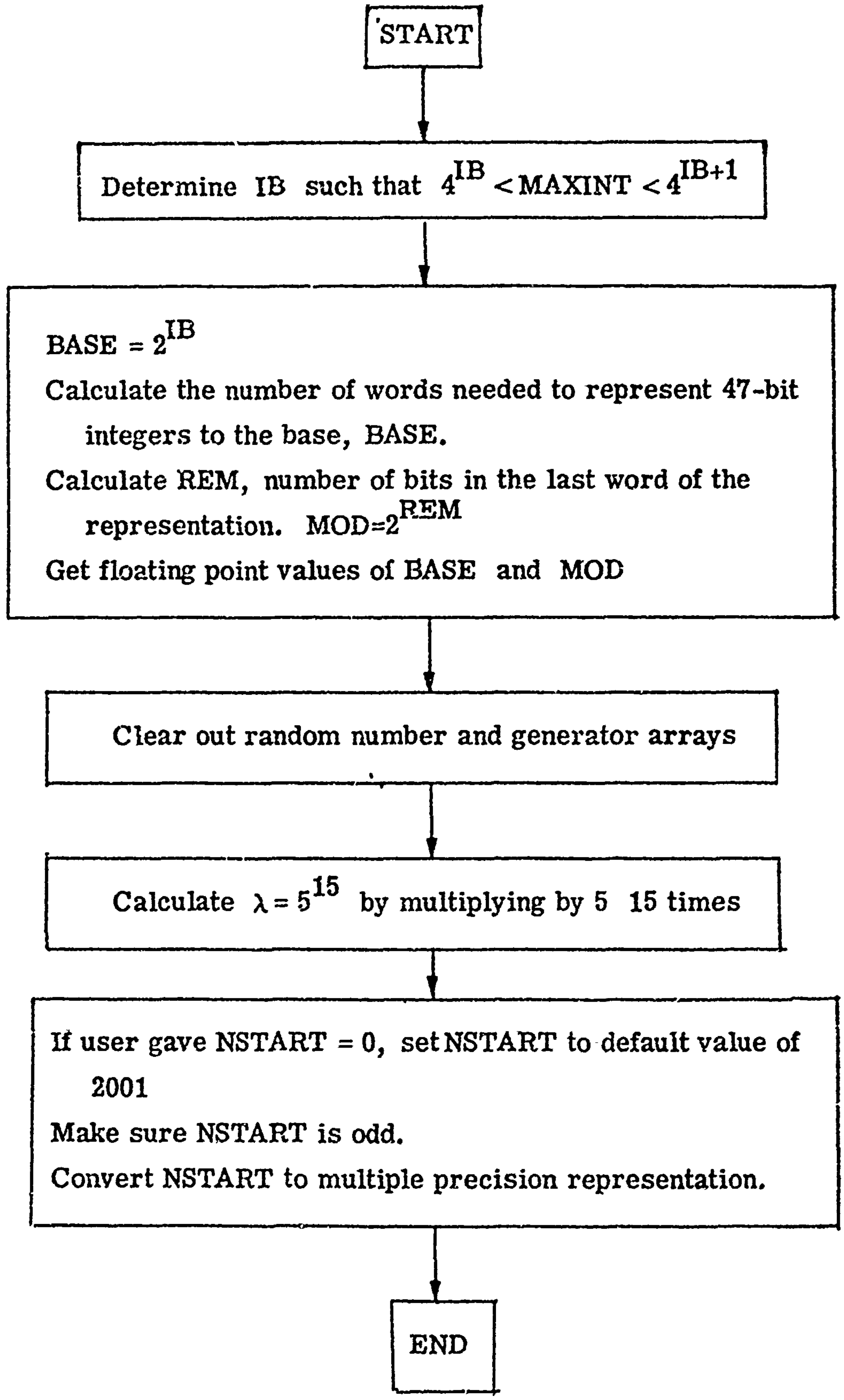



Figure A-4. Logic flow chart for RANSET

| | | |
|---|---|---|
| GEN(10) | - | An array containing the generator $\lambda(= 5^{15})$ in multiple precision representation |
| NWRD | - | The number of words used in the multiple precision representation of an integer |
| BASE | - | The base used in the multiple precision representation |
| MOD | - | The maximum value of the highest order 'digit' in the multiple precision representation |
| FBASE | - | Floating point value of BASE |
| FMOD | - | Floating point value of MOD |

RAN, GEN, NWRD, and NBASE are Fortran integers; FBASE and FMOD are Fortran real quantities.

An alternative method (unfortunately, not machine independent) of giving the routine a starting value is to save the array RAN at the end of a run and to restore RAN at the start of the new run (just after the RANSET call).

In the last step of the URAND flow the objective is conversion of the multiple precision integer random number R to a floating point random number X between 0 and 1. The multiple precision integer produced by the random number algorithm is represented by the 'digits' $r_1, r_2, \ldots\ldots, r_n$ (remember that $r_1$ is the lowest order digit. Thus,

$$R = r_1 + (\mathrm{BASE})\cdot r_2 + (\mathrm{BASE})^2 \cdot r_3 + \ldots\ldots + (\mathrm{BASE})^{N-1} \cdot r_N \ .$$

Notice that we have, from the manner in which N and MOD were established,

$$P = (\mathrm{BASE})^{N-1} \cdot \mathrm{MOD} \ .$$

The uniform random number desired is given by R/P. Thus we have,

$$X = \frac{R}{P} = \frac{r_1}{(BASE)^{N-1} \cdot MOD} + \frac{r_2}{(BASE)^{N-2} \cdot MOD} + \frac{r_3}{(BASE)^{N-3} \cdot MOD}$$

$$+ \ldots + \frac{r_{N-1}}{BASE \cdot MOD} + \frac{r_N}{MOD}$$

$$= \frac{1}{MOD}(r_N + \frac{1}{BASE}(r_{N-1} + \ldots \frac{1}{BASE}(r_2 + \frac{1}{BASE} \cdot r_1) \ldots))$$

Starting from the right it is easy to compute this iteratively.

## A. 6 FIRST 100 RANDOM NUMBERS PRODUCED BY MIRAN

For checkout purposes, Table A-1 lists the first 100 random numbers produced by MIRAN when the default value of NSTART, 2001, is used as the starting random number.

TABLE A.1

First 100 Random Numbers Produced by Machine-Independent Random Number Generator

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| .4338977 | .7858570 | .9904279 | .8066061 | .9646561 | .4108350 | .5079340 | .2304957 | .8253787 | .6423580 |
| .3724977 | .2999164 | .5649909 | .6669998 | .8994935 | .0315737 | .6045930 | .2886992 | .7004357 | .0770901 |
| .7623080 | .8476278 | .5594711 | .2497678 | .2726822 | .1385819 | .1319806 | .8203205 | .2988024 | .9646986 |
| .1032452 | .2193750 | .0736062 | .0596241 | .4543268 | .6998527 | .5666641 | .6486428 | .6572630 | .4945283 |
| .6556168 | .2540241 | .4256103 | .1514480 | .6868500 | .9272675 | .4724877 | .6947799 | .4599674 | .4813375 |
| .7548023 | .6871775 | .9435987 | .1946915 | .2781693 | .7060840 | .2893777 | .6662300 | .9024574 | .3585868 |
| .6711822 | .1615501 | .0777647 | .5482782 | .7349975 | .0763710 | .0634225 | .0418916 | .6368102 | .6480426 |
| .6594950 | .7903770 | .6175984 | .8909496 | .3714798 | .1235697 | .0322795 | .4017063 | .7728325 | .0014033 |
| .3249464 | .5456600 | .6429544 | .4225344 | .3104638 | .6921146 | .2768275 | .3440949 | .9040403 | .5348431 |
| .2713845 | .6928810 | .3603209 | .2550091 | .2913790 | .0058372 | .9934161 | .4184589 | .9584109 | .8090146 |

## REFERENCES


1. McGrath, E.J., et al., "Techniques for Efficient Monte Carlo Simulation - Selecting Probability Distribution", Vol. 1, SAI-72-409-LJ, March 1973.

2. McGrath, E.J., and D.C. Irving, "Techniques for Efficient Monte Carlo Simulations - Random Number Generation for Selected Probability Distributions", Vol. II, SAI-72-509-LJ, March 1973.

3. McGrath, E.J., and D.C. Irving, "Techniques for Efficient Monte Carlo Simulations - Variance Reduction", Vol. III, SAI-72-509-LJ, March 1973.

4. "APAIR MOD2, ASW Program Air Engagement Model", SAO Report 69-10, June 1969.

5. "APAIR MOD 26 ASW Program Air Engagement Model", Vol. I, User's Manual, SAO Technical Memorandum 71-12, July 1971. AD890139.

6. "APAIR MOD 2 ASW Program Air Engagement Model", Vol. II, Programmer's Manual, SAO Report 69-10, June 1969. AD0860262L.

7. [redacted]

8. "APSURF MOD 1, ASW Programs Surface Ship Engagement Model", Vol. 1, SAO Report No. 69-16 and 881385L.

9. Coveyou, R.R., and R.D. MacPherson, "Fourier Analysis of Uniform Random Number Generators", Journal of The Assoc. for Comput. Mach., 14(1): 100-119, January 1967.

10. Breipohl, A.M., "Probabilistic Systems Analysis", John Wiley and Sons, 1970.

11. Marsaglia, G., and T.A. Bray, "A Convenient Method For Generating Normal Variables", SIAM Review, 6, 1964.


Preceding page blank